User Guide

Amazon DataZone

Amazon DataZone: User Guide

Table of Contents

What is Amazon DataZone?

Amazon DataZone is a data management service that makes it faster and easier for you to catalog, discover, share, and govern data stored across AWS, on-premises, and third-party sources. With Amazon DataZone, administrators who oversee organization's data assets can manage and govern access to data using fine-grained controls. These controls help ensure access with the right level of privileges and context. Amazon DataZone makes it easy for engineers, data scientists, product managers, analysts, and business users to share and access data throughout an organization so they can discover, use, and collaborate to derive data-driven insights.

Amazon DataZone helps you deliver data to end users directly and simplifies your architecture by integrating data management services, including Amazon Redshift, Amazon Athena, Amazon QuickSight, AWS Glue, AWS Lake Formation, on-premises sources, third-party sources, and more.

Topics

What Can I Do with Amazon DataZone?

With Amazon DataZone, you can do the following:

- **Govern data access across organizational boundaries.** With Amazon DataZone, you can help ensure that the right data is accessed by the right user for the right purpose, in accordance with your organization's security regulations, without relying on individual credentials. You can also provide transparency on data asset usage and approve data subscriptions with a governed workflow. You can also monitor data assets across projects through usage auditing capabilities.
- **Connect data workers through shared data and tools to drive business insights.** With Amazon DataZone, you can increase business team's efficiency by collaborating seamlessly across teams and providing self-service access to data and analytics tools. You can use business terms to search, share, and access cataloged data stored in AWS, on-premises, or with third-party providers. And you can learn more about the data that you want to use by using Amazon DataZone business glossaries.

- **Automate data discovery and cataloging with machine learning.** With Amazon DataZone, you can reduce the time spent on manual entry of data attributes into the business data catalog. Richer data in the data catalog also improves the searching experience.

How Amazon DataZone supports and integrates with other AWS services?

Amazon DataZone supports three types of integrations with other AWS services:

- Producer data sources - you can publish data assets to the Amazon DataZone catalog from the data stored in AWS Glue Data Catalog and Amazon Redshift tables and views. You can also manually publish objects from Amazon Simple Storage Service (S3) to the Amazon DataZone catalog.
- Consumer tools - you can use Amazon Athena or Amazon Redshift query editors to access and analyze your data assets.
- Access control and fulfillment - Amazon DataZone supports granting access to AWS Lake Formation managed AWS Glue tables and Amazon Redshift tables and views. For all other data assets, Amazon DataZone publishes standard events related to your actions (e.g., approval given to a subscription request) to Amazon EventBridge. You can use these standard events to integrate with other AWS services or third-party solutions for custom integrations.

How can I access Amazon DataZone?

You can access Amazon DataZone in any of the following ways:

- Amazon DataZone console

  You can use the Amazon DataZone management console to access and configure your Amazon DataZone domains, blueprints, and users. For more information, see https://console.aws.amazon.com/datazone. The Amazon DataZone management console is also used to create the Amazon DataZone data portal.
- Amazon DataZone data portal

  The Amazon DataZone data portal is a browser-based web application where you can catalog, discover, govern, share, and analyze data in a self-service fashion. The data portal can authenticate you with credentials from your identity provider through AWS IAM Identity Center (successor to AWS SSO), or with your IAM credentials. You can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone.

- Amazon DataZone HTTPS API

  You can access Amazon DataZone programmatically by using the Amazon DataZone HTTPS API, which lets you issue HTTPS requests directly to the service. For more information, see the Amazon DataZone API Reference.

Amazon DataZone terminology and concepts

As you get started with Amazon DataZone, it is important that you understand its key concepts, terminology, and components.

Topics

Amazon DataZone components

Amazon DataZone includes the following four main components:

- Business data catalog - you can use this component to catalog data across your organization with business context and thus enable everyone in your organization to find and understand data quickly.
- Publish and subscribe workflows - you can use these automated workflows to secure data between producers and consumers in a self-service manner and to ensure that everyone in your organization has access to the right data for the right purpose.
- Projects and environments
  - In Amazon DataZone projects are business use case–based groupings of people, assets (data), and tools used to simplify access to the AWS analytics. Projects provide areas where project members can collaborate, exchange data, and share assets. By default, projects are configured so that only those who are explicitly added to the project are able to access the data and analytics tools within them. Projects manage the ownership of assets produced in accordance with project policies for data consumers to access.
  - Within Amazon DataZone projects, environments are collections of zero or more configured resources (for example, an Amazon S3 bucket, an AWS Glue database, or Amazon Athena

workgroup) on which a given set of IAM principals (for example, users with a contributor permissions) can operate.

- Data portal (outside the AWS Management Console) - this is a browser-based web application where different users can go to catalog, discover, govern, share, and analyze data in a self-service fashion. The data portal authenticates users with IAM credentials or existing credentials from your identity provider through AWS IAM Identity Center.

What are Amazon DataZone domains?

You can use Amazon DataZone domains to organize your assets, users, and their projects. By associating additional AWS accounts with your Amazon DataZone domains, you can bring together your data sources. You can then publish assets from these data sources to your domain's catalog, with metadata forms and glossaries that improve metadata completeness and quality. You can also search and browse these assets to see what data is published in the domain. Additionally, you can join projects to collaborate with others users, subscribe to assets, and use project environments to access analytics tools, including Amazon Athena and Amazon Redshift. Amazon DataZone domains enable you with the flexibility to reflect the data and analytics needs of your organizational structure, whether it's creating a single Amazon DataZone domain for your enterprise or multiple Amazon DataZone domains for different business units.

What are Amazon DataZone projects and environments?

Amazon DataZone enables teams and analytics users to collaborate on projects by creating use-case based grouping of teams, tools, and data.

- In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and consuming data in the Amazon DataZone catalog. Project members consume assets from the Amazon DataZone catalog and produce new assets using one or more analytical workflows. Projects support the following activities within the data portal:
  - Project owners can add members with owner and contributor permissions
  - Project members can be SSO users, SSO groups, and IAM users
  - Project members can request subscription to the assets in the data catalog

    Subscription approvals are provided to the projects

- In a Amazon DataZone project, environments are collections of zero or more configured resources (for example, an Amazon S3, an AWS Glue database, or an Amazon Athena workgroup), with a given set of IAM principals who can operate on those resources. Environments are created by using environment profiles which are pre-configured sets of resources and blueprints that provide reusable templates for creating environments. Environment profiles define settings such as the AWS account or region in which environments are deployed.

What are Amazon DataZone blueprints?

A blueprint with which the environment is created defines what AWS tools and services (for example, AWS Glue or Amazon Redshift) members of the project to which the environment belongs can use as they work with assets in the Amazon DataZone catalog.

In the current release of Amazon DataZone, the following default blueprints are supported:

Blueprint name	Description	Resources created
Data Lake blueprint	Enables Amazon DataZone project members to launch Data Lake producer and consumer services within the environment. As a *consumer*, it enables Amazon DataZone project members to access a 'read only' copy of Lake Formation -managed assets directly in Amazon Athena and in other Lake Formation-supported query engines. As a *producer*, it enables Amazon DataZone project members to create new LakeFormation-managed tables using Amazon Athena	Provides users with the ability to create and query Lake Formation tables using Amazon Athena. Amazon Athena workgroup, AWS Glue database with 'read only' Lake Formation permissions, 'read only' IAM permissions, and access to Amazon S3 that is managed by the project. AWS Glue database with 'create' and 'grant' Lake Formation permissions, 'read' and 'write' IAM permissions, AWS Glue ETL (extract, transform, and load) with tagging.

Blueprint name	Description	Resources created
	and to publish them to the Amazon DataZone catalog.	
Data Warehouse blueprint	As a *consumer*, this blueprint enables Amazon DataZone project members to connect to their own Amazon Redshift clusters to query remote data stores and to create and store new data sets. As a *producer*, this blueprint enables Amazon DataZone project members to connect to their own Amazon Redshift clusters to query remote data stores, to create new datasets, and to publish them to the Amazon DataZone catalog.	Access to the Amazon Redshift query editor, 'read' access to the subscribed data sources from the Amazon DataZone catalog, the ability to create local assets in the configured Amazon Redshift cluster. Access to the Amazon Redshift query editor, 'read' access to the subscribed data sources from the Amazon DataZone catalog, the ability to create and publish assets from the configured Amazon Redshift cluster.

Blueprint name	Description	Resources created
Amazon Sagemaker blueprint	This blueprint help data producers and consumers to seamlessly switch to Amazon SageMaker to collaborate on machine learning (ML) projects while enforcing access governance to data and ML assets. With the new built-in integration between Amazon DataZone and Amazon SageMaker, data consumers and producers can streamline ML governance across infrastructure setup, collaborate on business initiatives, and easily govern data and ML assets.	You can create an Amazon SageMaker domain that can search, subscribe and publish data and ML assets in Amazon DataZone. Also can subscribe and publish to AWS Glue databases and lake formation as configured.

In addition to the built-in blueprints, you can also enable the custom AWS serivce blueprint that allows you to configure Amazon DataZone to use your own existing IAM roles and AWS services that you already have set up in your organization. For more information, see Working with custom AWS service blueprints in Amazon DataZone.

What are Amazon DataZone inventory and publishing workflows?

Creating project inventory assets

In order to use Amazon DataZone to catalog your data, you must first bring your data (assets) as inventory of your project in Amazon DataZone. Creating inventory for a project, makes the assets discoverable only to that project’s members. Project inventory assets are not available to all domain users in search/browse unless explicitly published. In the current release of Amazon DataZone, you can add assets to the project inventory in the following ways:

- Create and run data sources via the data portal or by using the Amazon DataZone APIs. In the current release of Amazon DataZone, you can create and run data sources for AWS Glue and Amazon Redshift. By creating and running AWS Glue or Amazon Redshift data sources, you create assets in a chosen project inventory and import their technical metadata from the source database tables or data warehouses as inventory into Amazon DataZone.
- Using APIs, you can create assets from the available system asset types (AWS Glue, Amazon Redshift, Amazon S3 objects) or from your custom asset types.
  - Create custom asset types in a project inventory by using the Amazon DataZone APIs. The custom asset types can include ML models, dashboards, on-premises tables, etc.
  - Create assets from these custom asset types using Amazon DataZone APIs.
- Manually create assets for S3 objects using the Amazon DataZone data portal.

Curating of your project inventory assets - after creating a project inventory, data owners can curate their inventory assets with the required business metadata by adding or updating business names (asset and schema), descriptions (asset and schema), read me, glossary terms (asset and schema), and metadata forms. You can do this via the data portal or by using the Amazon DataZone APIs. Each edit to your asset creates a new inventory version.

Publishing project inventory assets to the Amazon DataZone catalog

The next step of using Amazon DataZone to catalog your data, is to make your project's inventory assets discoverable by the domain users. You can do this by publishing the inventory assets to the Amazon DataZone catalog. Only the latest version of the inventory asset can be published to the catalog and only the latest published version is active in the discovery catalog. If an inventory asset is updated after it's been published into the Amazon DataZone catalog, you must explicitly publish it again in order for the latest version to be in the discovery catalog. In the current release of Amazon DataZone, you can publish your project inventory assets to the Amazon DataZone catalog in the following ways:

- Manually publish your project inventory assets to the Amazon DataZone catalog either via the data portal or by using the Amazon DataZone APIs.
- As part of creating or editing data sources, enable the optional **Publish your AWS Glue assets to the catalog** or **Publish your Amazon Redshift assets to the catalog** settings to be used during the scheduled or automated data source runs. When this setting is enabled, a data source run adds assets to your project's inventory and then also publishes the inventory assets to the Amazon DataZone catalog. Note that if you publish directly, the assets might not have any

business metadata and will be made directly discoverable to all domain users. You can use this setting on your data sources either via the data portal or by using the Amazon DataZone APIs.

What are Amazon DataZone subscription and fulfillment workflows?

Once your assets are published to the Amazon DataZone catalog, your domain users can discover these assets, request and gain access to these assets, and continue to use Amazon DataZone to govern, share, and analyze these assets.

Users request access to an asset by subscribing to that asset on behalf of a project. Once a subscription request is created, owners of the asset get a notification and can review the subscription request and decide whether they want to approve or reject it. If the subscription request is approved by the data owner, the subscribing project is granted access to that asset.

Once a subscription request is approved, Amazon DataZone begins a subscription fulfillment workflow that automatically adds the asset to all the applicable environments within the project by creating the necessary grants in AWS Lake Formation or Amazon Redshift. This enables the subscribing project members to query the asset using one of the query tools (Amazon Athena or Amazon Redshift query editor) in their environments.

Amazon DataZone can trigger this automated fulfillment logic only for managed assets (this includes AWS Glue tables and Amazon Redshift tables and views). For all other asset types (unmanaged assets), Amazon DataZone can't automatically trigger fulfillment but instead publishes an event in Amazon Eventbridge with all the necessary details in the event payload so that you can create the necessary grants outside of Amazon DataZone. Amazon DataZone also provides the `updateSubscriptionStatus` API that enables you to update the status of the subscription once it is fulfilled outside of Amazon DataZone so that Amazon DataZone can notify the project members that they can start consuming the asset.

The user personas of Amazon DataZone

The following are the primary Amazon DataZone user personas:

- Domain administrators who own setting up Amazon DataZone as the analytics platform for their organization.

In the context of Amazon DataZone, domain administrators install Amazon DataZone in AWS accounts, create Amazon DataZone domains, and configure AWS account associations and identity providers associations with Amazon DataZone domains. Domain administrators also use other AWS service consoles such as AWS Organization and Service Catalog to configure Amazon DataZone.

- Data users who are the main users of Amazon DataZone (asset publishers and subscribers) for their analytics and machine learning tasks.

  Data users include data analytics workers, data scientists, and system users who produce and consume data assets. In the context of Amazon DataZone, data users create and join projects and environments, subscribe and consume data assets with pre-configured analytics or machine learning tools, and publish output data assets back to the Amazon DataZone domain catalog to share with others.
- System developers who build custom infrastructure templates and integrate Amazon DataZone with internal catalogs or production systems.

  In the context of Amazon DataZone, system developers build environment blueprints (infrastructure templates) or Infrastructure-As-Code CI/CD pipeline as a Environment provider, data pipelines to promote data assets across environments, catalog sync and subscription grant fulfillment adapters to integrate with internal catalogs, or integrations between Amazon DataZone APIs and internal user interfaces or production systems if needed.
- Data governance officers who own the definitions and risks of organizational security, privacy and other compliance policies and who make sure that the usage of Amazon DataZone in their organizations is in compliance with these definitions.

Amazon DataZone terminology

Domain

An Amazon DataZone domain is the organizing entity for connecting together your assets, users, and their projects. With Amazon DataZone domains, you have the flexibility to reflect the data and analytics needs of your organizational structure, whether it's creating a single Amazon DataZone domain for your enterprise or multiple datazone; domains for different business units or teams.

Associated account

Associating your AWS accounts with Amazon DataZone domains enables you to publish data from these AWS accounts into the Amazon DataZone catalog and create Amazon DataZone projects to work with your data across multiple AWS accounts. Account association requests can only be initiated in AWS accounts that own a Amazon DataZone domain. Account association requests can only be accepted by the administrative users of the invited AWS accounts. Once an AWS account is associated with an Amazon DataZone domain, you can register your data sources such as AWS Glue catalog and Amazon Redshift in this account to this domain. Being associated also enables an AWS account to create Amazon DataZone projects and environments.

An AWS account can be associated with one or more Amazon DataZone domain.

Data source

In Amazon DataZone, you can use data sources to import technical metadata of assets (data) from the source databases or data warehouses into Amazon DataZone. In the current release of Amazon DataZone, you can create and run data sources for AWS Glue and Amazon Redshift. By creating a data source, you establish a connection between Amazon DataZone and the source (AWS Glue Data Catalog or Amazon Redshift Warehouse) which enables you to read technical metadata, including tables names, columns names, and data types. By creating a data source you also kick off the initial data source run that creates new or updates existing assets in Amazon DataZone. While creating a data source or after the data source is successfully created, you also have the option to specify a schedule for your data source runs.

Data source run

In Amazon DataZone, a data source run is a task that Amazon DataZone performs in order to create assets in project inventories and also optionally to publish project inventory assets to the Amazon DataZone catalog. Data source runs can be automated (kicked off when a data source is initially created) or scheduled or manual. Data selection criteria enables you to fine-tune the existing and future data sets to be ingested into project inventories or the Amazon DataZone catalog and the frequency of metadata updates to those inventory or catalog assets.

Subscription target

In Amazon DataZone, subscription targets enable you to access the data to which you have subscribed in your projects. A subscription target specifies the location (for example, a database or a schema) and the required permissions (for example, an IAM role) that Amazon DataZone can use to establish a connection with the source data and to create the necessary grants so that members of the Amazon DataZone project can start querying the data to which they have subscribed.

Subscription request

In Amazon DataZone, a subscription request is a process that an Amazon DataZone project must follow in order to be granted access to a specific asset. Subscription requests can be approved, rejected, revoked, or granted.

Asset

In Amazon DataZone, an asset is an entity that presents a single physical data object (for examples, a table, a dashboard, a file) or virtual data object (for example, a view).

Asset type

Asset types define how assets are represented in the Amazon DataZone catalog. An asset type defines the schema for a specific type of asset. When assets are created, they are validated against the schema defined by their asset type (by default, the latest version). When an asset update occurs, Amazon DataZone creates a new asset version and enables Amazon DataZone users to operate on all asset versions.

Business glossary

In Amazon DataZone, a business glossary is a collection of business terms that may be associated with assets. A business glossary helps ensure that the same terms and definitions are used across an organization throughout its various data analytics tasks.

The terms in a business glossary can be added to assets and columns to classify or enhance the identification of those attributes during search. Glossary can be selected as the value type for a field in a metadata form that is associated with an asset. When a particular term is selected as the value for an asset's metadata form field, users can search for the business glossary term and find the associated assets.

Metadata form type

A metadata form type is a template that defines the metadata that is collected and saved when assets are created as inventory or published in a Amazon DataZone domain. Metadata form types can be associated with a data asset. Metadata form types help domain administrators to define metadata forms needed for that domain such as compliance information, regulation information, or classifications. It enables domain administrators to customize additional metadata for their assets. Amazon DataZone has system metadata form types such as asset-common-details-form-type, column-business-metadata-form-type, glue-table-form-type, glue-view-form-type, redshift-table-form-type, redshift-view-form-type, s3-object-collection-form-type, subscription-terms-form-type, and suggestion-form-type.

Metadata form

In Amazon DataZone, metadata forms define the metadata that is collected and saved when assets are created as inventory or published in a Amazon DataZone domain. Metadata form definitions are created in the catalog domain by a domain administrator. A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types.

A domain administrator applies a metadata form to assets in their domain by adding the metadata form to their domain. Asset publishers then provide any optional and required field values in the metadata form.

Project

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve creating assets in project inventories and thus making them discoverable by all project members, and then publishing, discovering, subscribing to, and consuming assets in the Amazon DataZone catalog. Project members consume assets from the Amazon DataZone catalog and produce new assets using one or more analytical workflows. Project members can be owners or contributors. Project owners can add or remove other users as owners or contributors and they can modify or delete projects. Other restrictions on contributors can be defined with policies. When a user creates a project, they become the first owner of that project.

Environment

An environment is a collection of configured resources (for example, an Amazon S3 bucket, an AWS Glue database, or an Amazon Athena workgroup), with a given set of IAM principals (with assigned contributor permissions) who can operate on those resources. Each environment may also have user principals who are authorized to access the resources and get access to data via subscription and fulfillment. Environments are designed to store actionable links into AWS services and external IDEs and consoles. Members of the project can access services such as the Amazon Athena console and more via deep links configured within an environment. SSO users and IAM users from the project can be further scoped down to use/access specific environments.

Environment profile

In Amazon DataZone, an environment profile is a template that you can use to create environments. Environment profiles are created by using blueprints.

With environment profiles, domain administrators can wrap blueprints with preconfigured parameters, and then data workers can quickly create any number of new environments by selecting existing environment profiles and specifying names for the new environments. This enables data workers to efficiently manage their projects and environments while ensuring that they satisfy data governance policies enforced by their domain administrators.

Blueprint

A blueprint with which the environment is created defines what AWS tools and services (for example, AWS Glue or Amazon Redshift) members of the project to which the environment belongs can use as they work with assets in the Amazon DataZone catalog.

In the current release of Amazon DataZone the following blueprints are supported:

- Data lake blueprint
- Data warehouse blueprint
- Amazon Sagemaker blueprint
- Custom AWS service blueprint

User profile

A user profile represents Amazon DataZone users. Amazon DataZone supports both IAM roles and SSO identities to interact with the Amazon DataZone Management Console and the data portal for different purposes. Domain administrators use IAM roles to perform the initial administrative domain-related work in the Amazon DataZone Management Console, including creating new Amazon DataZone domains, configuring metadata form types, and implementing policies. Data workers use their SSO corporate identities via Identity Center to log into the Amazon DataZone Data Portal and access projects where they have memberships.

Group profile

Group profiles represent groups of Amazon DataZone users. Groups can be manually created, or mapped to Active Directory groups of enterprise customers. In Amazon DataZone, groups serve two purposes. First, a group can map to a team of users in the organizational chart, and thus reduce the administrative work of a Amazon DataZone project owner when there are new employees joining or leaving a team. Second, corporate administrators use Active Directory groups to manage and update user statuses and so Amazon DataZone domain administrators can use these group memberships to implement Amazon DataZone domain policies.

Domain administrator

In Amazon DataZone, an IAM principal who creates an Amazon DataZone domain is the default domain administrator of that domain. Domain administrators in Amazon DataZone perform key functionalities for the domain, including creating domains, assigning other domain administrators, adding data sources and subscription targets, creating projects and environments, and assigning project owners.

Publisher

In Amazon DataZone, publishers publish assets into the Amazon DataZone catalog and can edit the metadata of the assets they publish. If granted this authority, publishers can approve or reject subscription requests to the assets they published in the Amazon DataZone catalog.

Subscriber

In Amazon DataZone, a subscriber is an Amazon DataZone project that wants to find, access, and consume assets in the Amazon DataZone catalog.

AWS account owner

In Amazon DataZone, AWS account owners create roles, policies, and permissions in their AWS accounts that enable these AWS accounts to be associated with Amazon DataZone domains.

What is new in Amazon DataZone?

This section describes new features and improvements in Amazon DataZone by release date.

Topics

- 2024
- 2023

2024

Amazon DataZone launches data lineage functionality

Released on 06/27/2024

Amazon DataZone launches data lineage in preview, helping customers visualize lineage events from OpenLineage-enabled systems or through API and trace data movement from source to consumption. Using Amazon DataZone's OpenLineage-compatible APIs, domain administrators and data producers can capture and store lineage events beyond what is available in Amazon DataZone, including transformations in Amazon S3, AWS Glue, and other services. Additionally, Amazon DataZone versions lineage with each event, enabling users to visualize lineage at any point in time or compare transformations across an asset's or job's history. This historical lineage provides a deeper understanding of how data has evolved, essential for troubleshooting, auditing, and validating the integrity of data assets. For more information, see Data lineage in Amazon DataZone (Preview)

Amazon DataZone launches custom AWS service blueprints

Released on 06/17/2024

With custom AWS service blueprints, if you have existing AWS resources including IAM roles, data lakes, data meshes, Amazon S3 buckets, and Amazon Redshift clusters, you are now able to specify permissions to these existing resources using your own custom IAM role, so that your Amazon DataZone users can leverage publication and subscription to share and govern these resources. With custom AWS service blueprints, Amazon DataZone administrators can configure AWS service environments using their own custom roles. They can configure actions links for these AWS service environments and thus provide federated access to any of their existing AWS resources. They can also configure subscription targets and data sources in these custom AWS service environments.

Administrators can set up AWS service environments in their own Amazon DataZone domain account or in any associated accounts from which they want to publish, subscribe to, discover, or govern data. For more information, see Working with custom AWS service blueprints in Amazon DataZone.

Enhancements to the data source creation flow

Released on 06/10/2024

Amazon DataZone has added enhancements to the data source creation flow to simplify access management for data producers. With these updates, when a data producer creates a data source for publishing their AWS Glue and Amazon Redshift assets, Amazon DataZone grants read-only permissions to the project members. When creating an AWS Glue data source, Amazon DataZone automatically grants 'read-only' permissions to the IAM role of the environment used to create the data source, allowing access to all tables in the associated AWS Glue databases. Similarly, for Amazon Redshift data sources, Amazon DataZone grants 'read-only' access to all tables in the Amazon Redshift schemas used in the data source. For more information, see Create and run an Amazon DataZone data source for the AWS Glue Data Catalog and Create and run an Amazon DataZone data source for Amazon Redshift.

Amazon DataZone launches integration with Amazon SageMaker

Released on 05/06/2024

Amazon DataZone launches integration with Amazon SageMaker to help data producers and consumers to seamlessly switch to Amazon SageMaker to collaborate on machine learning (ML) projects while enforcing access governance to data and ML assets. With the new built-in integration between Amazon DataZone and Amazon SageMaker, data consumers and producers can streamline ML governance across infrastructure setup, collaborate on business initiatives, and easily govern data and ML assets. For more information, see Working with the Amazon DataZone built-in blueprints and Working with associated accounts to publish and consume data.

Amazon DataZone launches integration with AWS Lake Formation hybrid access mode

Released on 04/03/2024

Amazon DataZone has introduced an integration with AWS Lake Formation hybrid access mode. This integration enables you to easily publish and share your AWS Glue tables through Amazon

DataZone, without the need to register them in AWS Lake Formation first. To get started, administrators enable the data location registration setting under the `DefaultDataLake` blueprint in the Amazon DataZone console. Then, when a data consumer subscribes to an AWS Glue table managed through IAM permissions, Amazon DataZone first registers the Amazon S3 locations of this table in hybrid mode, and then grants access to the data consumer by managing permissions on the table through AWS Lake Formation. This ensures that IAM permissions on the table continue to exist with newly-granted AWS Lake Formation permissions, without disrupting any existing workflows. For more information, see the Amazon DataZone integration with AWS Lake Formation hybrid mode .

Amazon DataZone launches integration with AWS Glue Data Quality

Released on 04/03/2024

Amazon DataZone launches integration with AWS Glue Data Quality and offers APIs to integrate data quality metrics from third-party data quality solutions. The new integration enables you to auto-publish AWS Glue Data Quality scores into the Amazon DataZone business data catalog. Amazon DataZone APIs can be used to ingest quality metrics from third-party sources. Once published, data consumers can easily search for data assets, view granular quality metrics, and identify failed checks and rules - empowering business decisions. For more information, see the Data quality in Amazon DataZone.

General availability release of AI recommendations for descriptions in Amazon DataZone

Released on 03/27/2024

Amazon DataZone announced the general availability release of the new generative AI-based capability to improve data discovery, data understanding and data usage by enriching the business data catalog. With a single click, data producers can generate comprehensive business data descriptions and context, highlight impactful columns, and include recommendations on analytical use cases. The launch adds support for APIs that data producers can use to programmatically generate descriptions for assets. For more information, see Using machine learning and generative AI.

Amazon DataZone launches enhancements to Amazon Redshift integration

Released on 03/21/2024

Amazon DataZone has introduced several enhancements to its Amazon Redshift integration, simplifying the process of publishing and subscribing to Amazon Redshift tables and views. These updates streamline the experience for both data producers and consumers, allowing them to quickly create data warehouse environments using pre-configured credentials and connection parameters provided by their Amazon DataZone administrators. Additionally, these enhancements grant administrators greater control over who can use the resources within their AWS accounts and Amazon Redshift clusters, and for what purpose.

- **Blueprint configuration**: once you enable the `DefaultDataWarehouseBlueprint` blueprint, you can control which projects can use the `DefaultDataWarehouseBlueprint` blueprint in your account to create environment profiles by assigning managing projects to the enabled blueprint. You can also create parameter sets on top of `DefaultDataWarehouseBlueprint` by providing parameters such as cluster, database, and an AWS Secret. You can also create AWS Secrets from within the Amazon DataZone console.
- **Environment profile**: when creating an environment profile, you can choose to provide your own Amazon Redshift parameters or use one of the parameter sets from the blueprint configuration. If you choose to use the parameter set created in the blueprint configuration, the AWS secret only requires `AmazonDataZoneDomain` tag (`AmazonDataZoneProject` tag is only required if you choose to provide your own parameter sets in the environment profile). In the environment profile, you can specify a list of authorized projects. Only authorized projects can use this environment profile to create data warehouse environments. You can also specify what data authorized projects are allowed to publish. Currently you can choose one of the following options: 1) Publish from any schema, 2) Publish from the default environment schema, 3) Don't allow publishing.
- **Environment**: Data producers or consumers can now select an environment profile to create environments, without the need to provide their own Amazon Redshift parameters including AWS Secret, cluster, workgroup, and database. These parameters are ported over to the environment from the environment profile. Along with the environment creation, Amazon DataZone now also creates default schema for the environment. Members of the project have read and write access to this schema and can easily publish any tables created in this schema to the catalog by running the default data source created as part of environment creation. Amazon Redshift parameters used to create environment can also be used for creating new data sources (instead of data producer to provide their own parameters in the data source creation).

AWS Cloud Formation Support for Amazon DataZone

Released on 01/18/2024

Users of Amazon DataZone can now leverage AWS CloudFormation to effectively model and manage a suite of Amazon DataZone resources. This approach facilitates consistent provisioning of resources, while also enabling lifecycle management through infrastructure as code practices. With custom templates, you can precisely define your required resources and their interdependencies. For more information, see the Amazon DataZone resource type reference.

Add IAM principals directly as members of Amazon DataZone projects

Released on 01/05/2024

You can now add IAM principals as project members, even if those IAM principals have not yet logged into Amazon DataZone (previous requirement). After a domain administrator or IT administrator adds `iam:GetUser` and `iam:GetRole` to the domain's domain execution role, project owners can add IAM principals as members simply by providing the Amazon Resouce Name (ARN) of the IAM role or IAM user. The IAM principal still must have the IAM permissions required to access Amazon DataZone and those can be configured in the IAM console. For more information, see Add members to a project.

Support for custom asset types from the Data Portal

Released on 01/05/2024

The support for custom assets enables Amazon DataZone to catalog assets via the Data Portal for unstructured data, including dashboards, queries, and models, making it easier for you to add custom assets directly in the data portal along with the previously available API support. The ability to create, update and publish custom assets in Amazon DataZone, enables you to share, find, subscribe to any type of asset and build a business workflow that provides governance of those assets. For more information, see Create custom asset types.

2023

Delete domain

Released on 12/27/2023

This is a feature that enables you to more easily delete your domains. Now, you can proceed with domain deletion even if it's not empty (as in contains projects, environments, assets, data sources, etc.). For more information, see Delete domains.

Hybrid mode

Released on 12/22/2023

Amazon DataZone has added support for the AWS Lake Formation hybrid mode. With this support, if you publish an AWS Glue table to Amazon DataZone with its AWS S3 location registered in Lake Formation under hybrid mode, Amazon DataZone treats this table as a managed assets and can manage the subscription grants to this table. Prior to this feature release, Amazon DataZone would treat this table as an unmanaged asset i.e., Amazon DataZone would not be able to grant subscriptions to this table. For more information, see Configure Lake Formation permissions for Amazon DataZone.

HIPAA eligibility

Released on 12/14/2023

Amazon DataZone is now U.S. Health Insurance Portability and Accountability Act of 1996 (HIPAA) compliant. To view the list of AWS services with HIPAA compliance see https://aws.amazon.com/compliance/hipaa-eligible-services-reference/.

AI recommendations for descriptions in Amazon DataZone (Preview)

Released on 11/28/2023

AWS announces the preview of a new generative AI-based capability in Amazon DataZone to improve data discovery, data understanding, and data usage by enriching the business data catalog. With a single click, data producers can generate comprehensive business data descriptions and context, highlight impactful columns, and include recommendations on analytical use cases. With AI recommendations for descriptions in Amazon DataZone, data consumers can identify data tables and columns required for analysis, which enhances data discoverability and cuts down on back-and-forth communications with data producers. The preview is available in Amazon DataZone domains provisioned in the following AWS Regions: US East (N. Virginia), US West (Oregon). For more information, see Using machine learning and generative AI.

DefaultDataLake blueprint enhancement

Released on 11/20/2023

Amazon DataZone has added an enhancement to the DefaultDataLake blueprint that provides you with better control over who can publish what data from your AWS account. There are two key changes that were introduced with this feature launch.

- In the console, once you enable the DefaultDataLake blueprint, you can control which projects can use the DefaultDataLake blueprint in your account to create environment profiles by assigning managing projects to the enabled blueprint.
- The second change is in the portal. If you create an environment profile using the DefaultDataLake blueprint, you can also select the authorized projects that are allowed to use the environment profile for creating environments. By default, all projects are allowed to use the data lake environment profile, but you can restrict the environment profile to specific projects and also control what data can be published using the environments created with the profile.

For more information, see Create an environment profile.

Setting up

To set up the Amazon DataZone, you must have an AWS account and set up the required IAM policies and permissions for Amazon DataZone.

Once you've set up your Amazon DataZone permissions, it is recommended that you complete the steps in the Getting started section that takes you through creating the Amazon DataZone domain, obtaining the data portal URL, and the basic Amazon DataZone workflows for data producers and data consumers.

Topics

- Sign up for an AWS account
- Configure the IAM permissions required to use the Amazon DataZone management console
- Configure the IAM permissions required to use the Amazon DataZone data portal
- Setting up AWS IAM Identity Center for Amazon DataZone

Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

If you have an AWS organization, create an account:

1. Sign in to the AWS Management Console and open the Organizations console at https://console.aws.amazon.com/organizations/.
2. In the navigation pane, choose **AWS accounts**.
3. Choose **Add an AWS account**.
4. Choose **Create an AWS account** and provide the requested details. Choose **Create AWS account**.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to an administrative user, and use only the root user to perform tasks that require root user access.

Configure the IAM permissions required to use the Amazon DataZone management console

Any user, group or role that wants to use the Amazon DataZone management console, must have the required permissions.

Topics

- Attach required and optional policies to a user, group, or role for Amazon DataZone console access
- Create a custom policy for IAM permissions to enable the Amazon DataZone service console simplified role creation
- Create a custom policy for permissions to manage an account associated with an Amazon DataZone domain
- (Optional) Create a custom policy for AWS Identity Center permissions to enable single sign-on (SSO) for your domain
- (Optional) Create a custom policy for AWS Identity Center permissions to add and remove SSO user and SSO group access to your Amazon DataZone domain.
- (Optional) Add your IAM principal as a key user to create your Amazon DataZone domain with a customer-managed key from AWS Key Management Service (KMS)

Attach required and optional policies to a user, group, or role for Amazon DataZone console access

Complete the following procedure to attach the required and optional custom policies to a user, group, or a role. For more information, see AWS managed policies for Amazon DataZone.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Policies**.

3. Choose the following policies to attach to your user, group, or a role.

   - In the list of policies, select the check box next to the **AmazonDataZoneFullAccess**. You can use the **Filter** menu and the search box to filter the list of policies. For more information, see AWS managed policy: AmazonDataZoneFullAccess.
   - (Optional) Create a custom policy for IAM permissions to enable the Amazon DataZone service console simplified role creation.
   - (Optional) Create a custom policy for AWS Identity Center permissions to enable single sign-on (SSO) for your domain.
   - (Optional) Create a custom policy for AWS Identity Center permissions to add and remove SSO user and SSO group access to your Amazon DataZone domain.

4. Choose **Actions**, and then choose **Attach**.

5. Choose the user, group, or role to which you want to attach the policy. You can use the **Filter** menu and the search box to filter the list of principal entities. After choosing the user, group, or role, choose **Attach policy**.

Create a custom policy for IAM permissions to enable the Amazon DataZone service console simplified role creation

Complete the following procedure to create a custom inline policy to have the necessary permissions to enable Amazon DataZone to create the necessary roles in the AWS management console on your behalf.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.

2. In the navigation pane, choose **Users** or **User groups**.

3. In the list, choose the name of the user or group to embed a policy in.

4. Choose the **Permissions** tab and, if necessary, expand the **Permissions policies** section.

5. Choose **Add permissions** and **Create inline policy** link.

6. On the **Create Policy** screen, in the **Policy editor** section, choose **JSON**.

   Create a policy document with the following JSON statements, and then choose **Next**.

   ```
   {
   ```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreatePolicy",
                "iam:CreateRole"
            ],
            "Resource": [
                "arn:aws:iam::*:policy/service-role/AmazonDataZone*",
                "arn:aws:iam::*:role/service-role/AmazonDataZone*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "iam:AttachRolePolicy",
            "Resource": "arn:aws:iam::*:role/service-role/AmazonDataZone*",
            "Condition": {
                "ArnLike": {
                    "iam:PolicyARN": [
                        "arn:aws:iam::aws:policy/AmazonDataZone*",
                        "arn:aws:iam::*:policy/service-role/AmazonDataZone*"
                    ]
                }
            }
        }
    ]
}
```

7. On the **Review policy** screen, enter a name for the policy. When you're satisfied with the policy, choose **Create policy**. Ensure that no errors appear in a red box at the top of the screen. Correct any that are reported.

Create a custom policy for permissions to manage an account associated with an Amazon DataZone domain

Complete the following procedure to create a custom inline policy to have the necessary permissions in an associated AWS account to list, accept, and reject resource shares of a domain, and then enable, configure, and disable environment blueprints in the associated account. To enable the optional Amazon DataZone service console simplified role creation available during blueprint configuration, you must also Create a custom policy for IAM permissions to enable the Amazon DataZone service console simplified role creation .

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users** or **User groups**.
3. In the list, choose the name of the user or group to embed a policy in.
4. Choose the **Permissions** tab and, if necessary, expand the P**ermissions policies** section.
5. Choose **Add permissions** and **Create inline policy** link.
6. On the **Create Policy** screen, in the **Policy editor** section, choose **JSON**. Create a policy document with the following JSON statements, and then choose **Next**.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "datazone:ListEnvironmentBlueprintConfigurations",
                "datazone:PutEnvironmentBlueprintConfiguration",
                "datazone:GetDomain",
                "datazone:ListDomains",
                "datazone:GetEnvironmentBlueprintConfiguration",
                "datazone:ListEnvironmentBlueprints",
                "datazone:GetEnvironmentBlueprint",
                "datazone:ListAccountEnvironments",
                "datazone:DeleteEnvironmentBlueprintConfiguration"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/AmazonDataZone",
                "arn:aws:iam::*:role/service-role/AmazonDataZone*"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:passedToService": "datazone.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "iam:AttachRolePolicy",
            "Resource": "arn:aws:iam::*:role/service-role/AmazonDataZone*",
            "Condition": {
                "ArnLike": {
                    "iam:PolicyARN": [
                        "arn:aws:iam::aws:policy/AmazonDataZone*",
                        "arn:aws:iam::*:policy/service-role/AmazonDataZone*"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "iam:ListRoles",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreatePolicy",
                "iam:CreateRole"
            ],
            "Resource": [
                "arn:aws:iam::*:policy/service-role/AmazonDataZone*",
                "arn:aws:iam::*:role/service-role/AmazonDataZone*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ram:AcceptResourceShareInvitation",
                "ram:RejectResourceShareInvitation",
                "ram:GetResourceShareInvitations"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "s3:CreateBucket",
            "Resource": "arn:aws:s3:::amazon-datazone*"
        }
    ]
}
```

7. On the **Review policy** screen, enter a name for the policy. When you're satisfied with the policy, choose **Create policy**. Ensure that no errors appear in a red box at the top of the screen. Correct any that are reported.

(Optional) Create a custom policy for AWS Identity Center permissions to enable single sign-on (SSO) for your domain

Complete the following procedure to create a custom inline policy to have the necessary permissions to enable single sign-on (SSO) using the AWS IAM Identity Center in Amazon DataZone.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users** or **User groups**.
3. In the list, choose the name of the user or group to embed a policy in.
4. Choose the **Permissions** tab and, if necessary, expand the **Permissions policies** section.
5. Choose **Add permissions** and **Create inline policy**.
6. On the **Create Policy** screen, in the **Policy editor** section, choose **JSON**.

   Create a policy document with the following JSON statements, and then choose **Next**.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "sso:DeleteManagedApplicationInstance",
                "sso:CreateManagedApplicationInstance",
                "sso:PutApplicationAssignmentConfiguration"
            ],
            "Resource": "*"
        }
    ]
}
```

7. On the **Review policy** screen, enter a name for the policy. When you're satisfied with the policy, choose **Create policy**. Ensure that no errors appear in a red box at the top of the screen. Correct any that are reported.

(Optional) Create a custom policy for AWS Identity Center permissions to add and remove SSO user and SSO group access to your Amazon DataZone domain.

Complete the following procedure to create a custom inline policy to have the necessary permissions to add and remove SSO user and SSO group access to your Amazon DataZone domain.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users** or **User groups**.
3. In the list, choose the name of the user or group to embed a policy in.
4. Choose the **Permissions** tab and, if necessary, expand the **Permissions policies** section.
5. Choose **Add permissions** and **Create inline policy**.
6. On the **Create Policy** screen, in the **Policy editor** section, choose **JSON**.

   Create a policy document with the following JSON statements, and then choose **Next**.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sso:GetManagedApplicationInstance",
        "sso:ListProfiles",
        "sso:GetProfiles",
        "sso:AssociateProfile",
        "sso:DisassociateProfile",
        "sso:GetProfile"
      ],
      "Resource": "*"
    }
  ]
}
```

7. On the **Review policy** screen, enter a name for the policy. When you're satisfied with the policy, choose **Create policy**. Ensure that no errors appear in a red box at the top of the screen. Correct any that are reported.

(Optional) Add your IAM principal as a key user to create your Amazon DataZone domain with a customer-managed key from AWS Key Management Service (KMS)

Before you can optionally create your Amazon DataZone domain with a customer-managed key (CMK) from the AWS Key Management Service (KMS), complete the following procedure to make your IAM principal a user of your KMS key.

1. Sign in to the AWS Management Console and open the KMS console at https://console.aws.amazon.com/kms/.
2. To view the keys in your account that you create and manage, in the navigation pane choose **Customer managed keys**.
3. In the list of KMS keys, choose the alias or key ID of the KMS key that you want to examine.
4. To add or remove key users, and to allow or disallow external AWS accounts to use the KMS key, use the controls in the **Key users** section of the page. Key users can use the KMS key in cryptographic operations, such as encrypting, decrypting, re-encrypting, and generating data keys.

Configure the IAM permissions required to use the Amazon DataZone data portal

Any user, group or role that wants to use the Amazon DataZone data portal or catalog must have the required permissions.

Topics

- Attach required policy to a user, group, or role for Amazon DataZone data portal access
- Attach required policy to a user, group, or role for Amazon DataZone catalog access
- Attach optional policy to a user, group, or role for Amazon DataZone data portal or catalog access if your domain is encrypted with a customer-managed key from AWS Key Management Service (KMS)

Attach required policy to a user, group, or role for Amazon DataZone data portal access

You can access the Amazon DataZone data portal by using either your AWS credentials or your single sign-on (SSO) credentials. Follow the instructions in the section below to set up the permissions required to access the data portal with your AWS credentials. For more information about using Amazon DataZone with SSO, see Setting up AWS IAM Identity Center for Amazon DataZone.

> **Note**
>
> Only IAM principals in your domain's AWS account can access the domain's data portal. IAM principals from other AWS accounts cannot access the domain's data portal.

Complete the following procedure to attach the required policy to a user, group, or a role. For more information, see AWS managed policies for Amazon DataZone.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users, User groups, or Roles**.
3. In the list, choose the name of the user, group, or role in which to embed a policy.

4. Choose the **Permissions** tab and, if necessary, expand the **Permissions policies** section.
5. Choose **Add permissions** and **Create inline policy** link.
6. On the **Create Policy** screen, in the Policy editor section, choose **JSON**. Create a policy document with the following JSON statements, and then choose **Next**.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "datazone:GetIamPortalLoginUrl"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

7. On the **Review policy** screen, enter a name for the policy. When you're satisfied with the policy, choose **Create policy**. Ensure that no errors appear in a red box at the top of the screen. Correct any that are reported.

Attach required policy to a user, group, or role for Amazon DataZone catalog access

Note

Only IAM principals in your domain's AWS account can access the domain's catalog. IAM principals from other AWS accounts cannot access the domain's catalog.

You can grant your IAM identities access to your Amazon DataZone domain's catalog via API and the SDK with the following procedure. If you want these IAM identities to also have access to the Amazon DataZone data portal, then additionally follow the procedure above to Attach required policy to a user, group, or role for Amazon DataZone data portal access. For more information, see AWS managed policies for Amazon DataZone.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Policies**.
3. In the list of policies, select the radio button next to the **AmazonDataZoneFullUserAccess** policy. You can use the **Filter** menu and the search box to filter the list of policies. For more information, see AWS managed policy: AmazonDataZoneFullUserAccess
4. Choose **Actions**, and then choose **Attach**.
5. Choose the user, group, or role to which you want to attach the policy by selecting the checkbox next to each principal. You can use the **Filter** menu and the search box to filter the list of principal entities. After choosing the user, group, or role, choose **Attach policy**.

Attach optional policy to a user, group, or role for Amazon DataZone data portal or catalog access if your domain is encrypted with a customer-managed key from AWS Key Management Service (KMS)

If you create your Amazon DataZone domain with your own KMS key for data encryption, you must also create an inline policy with the following permissions and attach it to your IAM principals so they can access the Amazon DataZone data portal or catalog.

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users, User groups, or Roles**.
3. In the list, choose the name of the user, group, or role in which to embed a policy.
4. Choose the **Permissions** tab and, if necessary, expand the **Permissions policies** section.
5. Choose **Add permissions** and **Create inline policy** link.
6. On the **Create Policy** screen, in the **Policy editor** section, choose **JSON**. Create a policy document with the following JSON statements, and then choose **Next**.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        }
    ]
}
```

7. On the **Review policy** screen, enter a name for the policy. When you're satisfied with the policy, choose **Create policy**. Ensure that no errors appear in a red box at the top of the screen. Correct any that are reported.

Setting up AWS IAM Identity Center for Amazon DataZone

> **Note**
>
> AWS Identity Center must be enabled in the same AWS Region as your Amazon DataZone domain. Currently, AWS Identity Center can only be enabled in a single AWS Region.

You can access the Amazon DataZone data portal by using either your single sign-on (SSO) credentials or AWS credentials. Follow the instructions in this section to set up AWS IAM Identity Center for Amazon DataZone. For more information about using Amazon DataZone with your AWS credentials, see Configure the IAM permissions required to use the Amazon DataZone management console.

You can skip the procedures in this section if you already have AWS IAM Identity Center (successor to AWS Single Sign-On) enabled and configured in the same AWS region where you want to create your Amazon DataZone domain.

Complete the following procedure to enable AWS IAM Identity Center (successor to AWS Single Sign-On).

1. To enable AWS IAM Identity Center, you must sign in to the AWS Management Console by using the credentials of your AWS Organizations management account. You can't enable IAM

Identity Center while signed in with credentials from an AWS Organizations member account. For more information, see Creating and managing an organization in the AWS Organizations User Guide.

2. Open the AWS IAM Identity Center (successor to AWS Single Sign-On) console and use the region selector in the top navigation bar to choose the AWS region in which you want create your Amazon DataZone domain.
3. Choose **Enable**.
4. Choose your identity source.

   By default, you get an IAM Identity Center store for quick and easy user management. Optionally, you can connect an external identity provider instead. In this procedure, we use the default IAM Identity Center store.

   For more information, see Choose your identity source.
5. In the IAM Identity Center navigation pane, choose **Groups**, and choose **Create group**. Enter the group name and choose **Create**.
6. In the IAM Identity Center navigation pane, choose **Users**.
7. On the **Add user** screen, enter the required information and choose **Send an email to the user with password setup instructions**. The user should get an email about the next setup steps.
8. Choose **Next: Groups**, choose the group that you want, and choose **Add user**. Users should receive an email inviting them to use SSO. In this email, they need to choose Accept invitation and set the password.

After you create your Amazon DataZone domain, you can enable AWS Identity Center for Amazon DataZone and provide access to your SSO users and SSO groups. For more information, see Enable IAM Identity Center for Amazon DataZone.

Getting started

The information in this section helps you get started using Amazon DataZone. If you are new to Amazon DataZone, start by becoming familiar with the concepts and terminology presented in Amazon DataZone terminology and concepts.

This getting started section takes you through the following Amazon DataZone quickstart workflows:

Topics

- Amazon DataZone quickstart with AWS Glue data
- Amazon DataZone quickstart with Amazon Redshift data
- Amazon DataZone quickstart with sample scripts

⚠ **Important**

Before you begin the steps in either of these quickstart workflows, you must complete the procedures described in the Setting Up section of this guide. If you are using a brand new AWS account, you must configure permissions required to use the Amazon DataZone management console. If you are using an AWS account that has existing AWS Glue Data Catalog objects, you must also configure Lake Formation permissions to Amazon DataZone.

Amazon DataZone quickstart with AWS Glue data

Topics

- Step 1 - Create the Amazon DataZone domain and data portal
- Step 2 - Create the publishing project
- Step 3 - Create the environment
- Step 4 - Produce data for publishing
- Step 5 - Gather metadata from AWS Glue
- Step 6 - Curate and publish the data asset
- Step 7 - Create the project for data analysis
- Step 8 - Create an environment for data analysis

- Step 9 - Search the data catalog and subscribe to data
- Step 10 - Approve the subscription request
- Step 11 - Build a query and analyze data in Amazon Athena

Step 1 - Create the Amazon DataZone domain and data portal

This section describes the steps of creating an Amazon DataZone domain and data portal for this workflow.

Complete the following procedure to create an Amazon DataZone domain. For more information about Amazon DataZone domains, see Amazon DataZone terminology and concepts.

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone, sign in, and then choose **Create domain**.

   Note

   If you want to use an existing Amazon DataZone domain for this workflow, choose **View domains**, then choose the domain that you want to use, and then proceed to Step 2 of creating a publishing project.

2. On the **Create domain** page, provide values for the following fields:

   - **Name** - specify a name for your domain. For the purposes of this workflow, you can call this domain **Marketing**.
   - **Description** - specify an optional domain description.
   - **Data encryption** - your data is encrypted by default with a key that AWS owns and manages for you. For this use case, you can leave the default data encryption settings.

     For more information about using customer managed keys, see Data encryption at rest for Amazon DataZone. If you use your own KMS key for data encryption, you must include the following statement in your default AmazonDataZoneDomainExecutionRole.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "*"
    }
  ]
}
```

- **Service access** - leave the selected by default **Use a default role** option unchanged.

> **Note**
>
> If you are using an existing Amazon DataZone domain for this workflow, you can choose **Use an existing service role** option and then choose an existing role from the drop-down menu.

- Under **Quick setup**, choose **Set up this account for data consumption and publishing**. This option enables the built-in Amazon DataZone blueprints of **Data lake** and **Data warehouse**, and configures the required permissions, resources, a default project, and default data lake and data warehouse environment profiles for this account. For more information about Amazon DataZone blueprints, see [Amazon DataZone terminology and concepts](#).
- Keep the remaining fields under **Permissions details** unchanged.

> **Note**
>
> If you have an existing Amazon DataZone domain, you can choose the **Use an existing service role** option and then choose an existing role from the drop-down menu for the **Glue Manage Access role**, **Redshift Manage Access role**, and **Provisioning role**.

- Keep the fields under **Tags** unchanged.
- Choose **Create domain**.

3. Once the domain is successfully created, choose this domain, and on the domain's summary page, note the **Data portal URL** for this domain. You can use this URL to access your Amazon DataZone data portal in order to complete the rest of the steps in this workflow. You can also navigate to the data portal by choosing **Open data portal**.

> **Note**
>
> In the current release of Amazon DataZone, once the domain is created, the URL generated for the data portal cannot be modified.

Domain creation can take several minutes to complete. Wait for the domain to have a status of **Available** before proceeding to the next step.

Step 2 - Create the publishing project

This section describes the steps required to create the publishing project for this workflow.

1. Once you complete Step 1 above and create a domain, you'll see the **Welcome to Amazon DataZone!** window. In this window, choose **Create project**.
2. Specify the project name, for example, for this workflow, you can name it **SalesDataPublishingProject**, then leave the rest of the fields unchanged, and then choose **Create**.

Step 3 - Create the environment

This section describes the steps required to create an environment for this workflow.

1. Once you complete Step 2 above and create your project, you'll see the **Your project is ready to use** window. In this window, choose **Create environment**.
2. On the **Create environment** page, specify the following and then choose **Create environment**.
3. Specify values for the following:
   - **Name** - specify the name for the environment. For this walkthrough, you can call it `Default data lake environment`.
   - **Description** - specify a description for the environment.
   - **Environment profile** - choose the **DataLakeProfile** environment profile. This enables you to use Amazon DataZone in this workflow to work with data in Amazon S3, AWS Glue Catalog, and Amazon Athena.
   - For this walkthrough, keep the rest of the fields unchanged.
4. Choose **Create environment**.

Step 4 - Produce data for publishing

This section describes the steps required to produce data for publishing in this workflow.

1. Once you complete step 3 above, in your `SalesDataPublishingProject` project, in the right-hand panel, under **Analytics tools**, choose **Amazon Athena**. This opens the Athena query editor using your project's credentials for authentication. Make sure that your publishing environment is selected in the **Amazon DataZone environment** dropdown and the `<environment_name>%_pub_db` database is selected as in the query editor.
2. For this walkthrough, you are using the **Create Table as Select** (CTAS) query script to create a new table that you want to publish to Amazon DataZone. In your query editor, execute this CTAS script to create a `mkt_sls_table` table that you can publish and make available for search and subscription.

   ```
   CREATE TABLE mkt_sls_table AS
   SELECT 146776932 AS ord_num, 23 AS sales_qty_sld, 23.4 AS wholesale_cost, 45.0 as
    lst_pr, 43.0 as sell_pr, 2.0 as disnt, 12 as ship_mode,13 as warehouse_id, 23 as
    item_id, 34 as ctlg_page, 232 as ship_cust_id, 4556 as bill_cust_id
   UNION ALL SELECT 46776931, 24, 24.4, 46, 44, 1, 14, 15, 24, 35, 222, 4551
   UNION ALL SELECT 46777394, 42, 43.4, 60, 50, 10, 30, 20, 27, 43, 241, 4565
   UNION ALL SELECT 46777831, 33, 40.4, 51, 46, 15, 16, 26, 33, 40, 234, 4563
   UNION ALL SELECT 46779160, 29, 26.4, 50, 61, 8, 31, 15, 36, 40, 242, 4562
   UNION ALL SELECT 46778595, 43, 28.4, 49, 47, 7, 28, 22, 27, 43, 224, 4555
   UNION ALL SELECT 46779482, 34, 33.4, 64, 44, 10, 17, 27, 43, 52, 222, 4556
   UNION ALL SELECT 46779650, 39, 37.4, 51, 62, 13, 31, 25, 31, 52, 224, 4551
   UNION ALL SELECT 46780524, 33, 40.4, 60, 53, 18, 32, 31, 31, 39, 232, 4563
   UNION ALL SELECT 46780634, 39, 35.4, 46, 44, 16, 33, 19, 31, 52, 242, 4557
   UNION ALL SELECT 46781887, 24, 30.4, 54, 62, 13, 18, 29, 24, 52, 223, 4561
   ```

   Make sure that the **mkt_sls_table** table is successfully created in the **Tables and views** section on the left-hand side. Now you have a data asset that can be published into the Amazon DataZone catalog.

Step 5 - Gather metadata from AWS Glue

This section describes the step of gathering metadata from AWS Glue for this workflow.

1. Once you complete step 4 above, in the Amazon DataZone data portal, choose the `SalesDataPublishingProject` project, then choose the **Data** tab, and then choose **Data sources** in the left-hand panel.
2. Choose the source that was created as part of the environment creation process.
3. Choose **Run** next to the **Action** dropdown menu and then choose the refresh button. Once the data source run is complete, the assets are added to the Amazon DataZone inventory.

Step 6 - Curate and publish the data asset

This section describes the steps of curating and publishing the data asset in this workflow.

1. Once you complete step 5 above, in the Amazon DataZone data portal, choose the `SalesDataPublishingProject` project that you created in the previous step, choose the **Data** tab, choose **Inventory data** in the left-hand panel, and locate the `mkt_sls_table` table.
2. Open `mkt_sls_table` asset's details page to see the automatically generated business names. Choose the **Automatically generated metadata** icon to view the auto-generated names for asset and columns. You can either accept or reject each name individually or choose **Accept all** to apply the generated names. Optionally, you can also add the available metadata form to your asset and select glossary terms to classify your data.
3. Choose **Publish asset** to publish the `mkt_sls_table` asset.

Step 7 - Create the project for data analysis

This section describes the steps of creating the project for data analysis. This is the beginning of the data consumer steps of this workflow.

1. Once you complete step 6 above, in the Amazon DataZone data portal, choose **Create project** from the **Project** drop-down menu.
2. On the **Create project** page, specify the project name, for example, for this workflow, you can name it **MarketingDataAnalysisProject**, then leave the rest of the fields unchanged, and then choose **Create**.

Step 8 - Create an environment for data analysis

This section describes the steps of creating an environment for data analysis.

1. Once you complete step 7 above, in the Amazon DataZone data portal, choose the `MarketingDataAnalysisProject` project, then choose the **Environments** tab, and then choose **Create environment**.
2. On the **Create environment** page, specify the following and then choose **Create environment**.

   - **Name** - specify the name for the environment. For this walkthrough, you can call it `Default data lake environment`.
   - **Description** - specify a description for the environment.
   - **Environment profile** - choose the built-in **DataLakeProfile** environment profile.
   - For this walkthrough, keep the rest of the fields unchanged.

Step 9 - Search the data catalog and subscribe to data

This section describes the steps of searching the data catalog and subscribing to data.

1. Once you complete step 8 above, in the Amazon DataZone data portal, choose the Amazon DataZone icon, and in the Amazon DataZone **Search** field, search for data assets using keywords (e.g., 'catalog' or 'sales') in the data portal's **Search** bar.

   If necessary, apply filters or sorting, and once you locate the **Product Sales Data** asset, you can choose it to open the asset's details page.
2. On the **Catalog Sales Data** asset's details page, choose **Subscribe**.
3. In the **Subscribe** dialog, choose your **MarketingDataAnalysisProject** consumer project from the dropdown, then specify the reason for your subscription request, and then choose **Subscribe**.

Step 10 - Approve the subscription request

This section describes the steps of approving the subscription request.

1. Once you complete step 9 above, in the Amazon DataZone data portal, choose the **SalesDataPublishingProject** project with which you published your asset.
2. Choose the **Data** tab, then **Published data**, and then chose **Incoming requests**.
3. Now you can see the row for the new request that needs an approval. Choose **View request**. Provide a reason for approval and choose **Approve**.

Step 11 - Build a query and analyze data in Amazon Athena

Now that you have successfully published an asset to the Amazon DataZone catalog and subscribed to it, you can analyze it.

1. In the Amazon DataZone data portal, choose your **MarketingDataAnalysisProject** consumer project and then, from the right-hand panel, under **Analytics tools**, choose the **Query data** link with Amazon Athena. This opens the Amazon Athena query editor using your project’s credentials for authentication. Choose the **MarketingDataAnalysisProject** consumer environment from the **Amazon DataZone Environment** dropdown in the query editor and then choose your project's `<environment_name>%sub_db` from the database dropdown.
2. You can now run queries on the subscribed table. You can choose the table from **Tables and Views**, and then choose **Preview** to have the select statement on the editor screen. Run the query to see the results.

Amazon DataZone quickstart with Amazon Redshift data

Topics

Step 1 - Create the Amazon DataZone domain and data portal

Complete the following procedure to create an Amazon DataZone domain. For more information about Amazon DataZone domains, see Amazon DataZone terminology and concepts.

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone, sign in, and then choose **Create domain**.

   > **Note**
   >
   > If you want to use an existing Amazon DataZone domain for this workflow, choose View domains, then choose the domain that you want to use, and then proceed to Step 2 of creating a publishing project.

2. On the **Create domain** page, provide values for the following fields:

   - **Name** - specify a name for your domain. For the purposes of this workflow, you can call this domain `Marketing`.
   - **Description** - specify an optional domain description.
   - **Data encryption** - your data is encrypted by default with a key that AWS owns and manages for you. For this walkthrough, you can leave the default data encryption settings.

     For more information about using customer managed keys, see Data encryption at rest for Amazon DataZone. If you use your own KMS key for data encryption, you must include the following statement in your default AmazonDataZoneDomainExecutionRole.

     ```
     {
       "Version": "2012-10-17",
       "Statement": [
         {
           "Effect": "Allow",
           "Action": [
             "kms:Decrypt",
             "kms:GenerateDataKey"
           ],
           "Resource": "*"
         }
       ]
     }
     ```

   - **Service access** - choose the **Use a custom service role** option and then choose the **AmazonDataZoneDomainExecutionRole** from the drop-down menu.

- Under **Quick setup**, choose **Set up this account for data consumption and publishing**. This option enables the built-in Amazon DataZone blueprints of **Data lake** and **Data warehouse**, and configures the required permissions and resources to complete the rest of the steps in this workflow. For more information about Amazon DataZone blueprints, see Amazon DataZone terminology and concepts.
- Keep the remaining fields under **Permissions details** and **Tags** unchanged and then choose **Create domain**.

3. Once the domain is successfully created, choose this domain, and on the domain's summary page, note the **Data portal URL** for this domain. You can use this URL to access your Amazon DataZone data portal in order to complete the rest of the steps in this workflow.

> **Note**
>
> In the current release of Amazon DataZone, once the domain is created, the URL generated for the data portal cannot be modified.

Domain creation can take several minutes to complete. Wait for the domain to have a status of **Available** before proceeding to the next step.

Step 2 - Create the publishing project

The following section describes the steps of creating the publishing project in this workflow.

1. Once you complete Step 1, navigate to the Amazon DataZone data portal using the data portal URL and log in using your single sign-on (SSO) or AWS IAM credentials.
2. Choose **Create project**, specify the project name, for example, for this workflow, you can name it **SalesDataPublishingProject**, then leave the rest of the fields unchanged, and then choose **Create**.

Step 3 - Create the environment

The following section describes the steps of creating an environment in this workflow.

1. Once you complete Step 2, in the Amazon DataZone data portal, choose the `SalesDataPublishingProject` project that you created in the previous step, then choose the **Environments** tab, and then choose **Create environment**.

2. On the **Create environment** page, specify the following and then choose **Create environment**.

    - **Name** - specify the name for the environment. For this walkthrough, you can call it `Default data warehouse environment`.
    - **Description** - specify a description for the environment.
    - **Environment profile** - choose the **DataWarehouseProfile** environment profile.
    - Provide the name of your Amazon Redshift cluster, database name, and the secret ARN for the Amazon Redshift cluster where your data is stored.

    > **Note**
    >
    > Make sure that your secret in AWS Secrets Manager includes the following tags (key/value):
    >
    > - For Amazon Redshift cluster - datazone.rs.cluster: <cluster_name:database name>
    >
    >   For Amazon Redshift Serverless workgroup - datazone.rs.workgroup: <workgroup_name:database_name>
    > - AmazonDataZoneProject: <projectID>
    > - AmazonDataZoneDomain: <domainID>
    >
    > For more information, see Storing database credentials in AWS Secrets Manager. The database user you provide in the AWS Secrets Manager must have super user permissions.

Step 4 - Produce data for publishing

The following section describes the steps of producing data for publishing in this workflow.

1. Once you complete Step 3, in the Amazon DataZone data portal, choose the `SalesDataPublishingProject` project, and then, in the right-hand panel, under **Analytics tools**, choose **Amazon Redshift**. This opens the Amazon Redshift query editor using your project's credentials for authentication.
2. For this walkthrough, you are using the **Create Table as Select** (CTAS) query script to create a new table that you want to publish to Amazon DataZone. In your query editor, execute this CTAS script to create a `mkt_sls_table` table that you can publish and make available for search and subscription.

```
CREATE TABLE mkt_sls_table AS
SELECT 146776932 AS ord_num, 23 AS sales_qty_sld, 23.4 AS wholesale_cost, 45.0 as
 lst_pr, 43.0 as sell_pr, 2.0 as disnt, 12 as ship_mode,13 as warehouse_id, 23 as
 item_id, 34 as ctlg_page, 232 as ship_cust_id, 4556 as bill_cust_id
UNION ALL SELECT 46776931, 24, 24.4, 46, 44, 1, 14, 15, 24, 35, 222, 4551
UNION ALL SELECT 46777394, 42, 43.4, 60, 50, 10, 30, 20, 27, 43, 241, 4565
UNION ALL SELECT 46777831, 33, 40.4, 51, 46, 15, 16, 26, 33, 40, 234, 4563
UNION ALL SELECT 46779160, 29, 26.4, 50, 61, 8, 31, 15, 36, 40, 242, 4562
UNION ALL SELECT 46778595, 43, 28.4, 49, 47, 7, 28, 22, 27, 43, 224, 4555
UNION ALL SELECT 46779482, 34, 33.4, 64, 44, 10, 17, 27, 43, 52, 222, 4556
UNION ALL SELECT 46779650, 39, 37.4, 51, 62, 13, 31, 25, 31, 52, 224, 4551
UNION ALL SELECT 46780524, 33, 40.4, 60, 53, 18, 32, 31, 31, 39, 232, 4563
UNION ALL SELECT 46780634, 39, 35.4, 46, 44, 16, 33, 19, 31, 52, 242, 4557
UNION ALL SELECT 46781887, 24, 30.4, 54, 62, 13, 18, 29, 24, 52, 223, 4561
```

Make sure that the **mkt_sls_table** table is successfully created. Now you have a data asset that can be published into the Amazon DataZone catalog.

Step 5 - Gather metadata from Amazon Redshift

The following section describes the steps of gathering metadata from Amazon Redshift.

1. Once you complete Step 4, in the Amazon DataZone data portal, choose the `SalesDataPublishingProject` project, then choose the **Data** tab, and then choose **Data sources**.
2. Choose the source that was created as part of the environment creation process.
3. Choose **Run** next to the **Action** dropdown menu and then choose the refresh button. Once the data source run is complete, the assets are added to the Amazon DataZone inventory.

Step 6 - Curate and publish the data asset

The following section describes the steps of curating and publishing the data asset in this workflow.

1. Once you complete step 5, in the Amazon DataZone data portal, choose the `SalesDataPublishingProject` project, then choose the **Data** tab, choose **Inventory data**, and locate the `mkt_sls_table` table.
2. Open `mkt_sls_table` asset's details page to see the automatically generated business names. Choose the **Automatically generated metadata** icon to view the auto-generated names for asset and columns. You can either accept or reject each name individually or choose **Accept all** to apply the generated names. Optionally, you can also add the available metadata form to your asset and select glossary terms to classify your data.
3. Choose **Publish** to publish the `mkt_sls_table` asset.

Step 7 - Create the project for data analysis

The following section describes the steps of creating te project for data analysis in this workflow.

1. Once you complete Step 6, in the Amazon DataZone data portal, choose **Create project**.
2. In the **Create project** page, specify the project name, for example, for this workflow, you can name it **MarketingDataAnalysisProject**, then leave the rest of the fields unchanged, and then choose **Create**.

Step 8 - Create an environment for data analysis

The following section describes the steps of creating an environment for data analysis in this workflow.

1. Once you complete Step 7, in the Amazon DataZone data portal, choose the `MarketingDataAnalysisProject` project that you created in the previous step, then choose the **Environments** tab, and then choose **Add environment**.
2. On the **Create environment** page, specify the following and then choose **Create environment**.

   - **Name** - specify the name for the environment. For this walkthrough, you can call it `Default data warehouse environment`.
   - **Description** - specify a description for the environment.
   - **Environment profile** - choose **DataWarehouseProfile** environment profile.
   - Provide the name of your Amazon Redshift cluster, database name, and the secret ARN for the Amazon Redshift cluster where your data is stored.

> **Note**
>
> Make sure that your secret in AWS Secrets Manager includes the following tags (key/value):
>
> - For Amazon Redshift cluster - datazone.rs.cluster: <cluster_name:database name>
>
>   For Amazon Redshift Serverless workgroup - datazone.rs.workgroup: <workgroup_name:database_name>
> - AmazonDataZoneProject: <projectID>
> - AmazonDataZoneDomain: <domainID>
>
> For more information, see Storing database credentials in AWS Secrets Manager.
> The database user you provide in the AWS Secrets Manager must have super user permissions.

- For this walkthrough, keep the rest of the fields unchanged.

Step 9 - Search the data catalog and subscribe to data

The following section describes the steps of searching the data catalog and subscribing to data.

1. Once you complete Step 8, in the Amazon DataZone data portal, search for data assets using keywords (e.g., 'catalog' or 'sales') in the data portal's **Search** bar.

   If necessary, apply filters or sorting, and once you locate the Product Sales Data asset, you can choose it to open the asset's details page.
2. On the Product Sales Data asset's details page, choose **Subscribe**.
3. In the dialog, choose your consumer project from the dropdown, provide the reason for access request, and then choose **Subscribe**.

Step 10 - Approve the subscription request

The following section describes the steps of approving the subscription request in this workflow.

1. Once you complete Step 9, in the Amazon DataZone data portal, choose the **SalesDataPublishingProject** project with which you published your asset.
2. Choose the **Data** tab, then **Published data**, and then **Incoming requests**.

3. Choose the view request link and then choose **Approve**.

Step 11 - Build a query and analyze data in Amazon Redshift

Now that you have successfully published an asset to the Amazon DataZone catalog and subscribed to it, you can analyze it.

1. In the Amazon DataZone data portal, on the right-hand panel, click the Amazon Redshift link. This opens the Amazon Redshift query editor using project's credential for authentication.
2. You can now run a query (select statement) on the subscribed table. You can click on the table (three-vertical-dots option) and choose preview to have select statement on the editor screen. Execute the query to see the results.

Amazon DataZone quickstart with sample scripts

The following section describes sample scripts that invoke various Amazon DataZone APIs that you can use to complete the following tasks:

Topics

Create an Amazon DataZone domain and data portal

You can use the following sample script to create an Amazon DataZone domain. For more information about Amazon DataZone domains, see Amazon DataZone terminology and concepts.

```
import sys
import boto3

// Initialize datazone client
region = 'us-east-1'
dzclient = boto3.client(service_name='datazone', region_name='us-east-1')

// Create DataZone domain
def create_domain(name):
    return dzclient.create_domain(
        name = name,
        description = "this is a description",
        domainExecutionRole = "arn:aws:iam::<account>:role/
AmazonDataZoneDomainExecutionRole",
    )
```

Create a publishing project

You can use the following sample script to create a publishing project in Amazon DataZone.

```
// Create Project
def create_project(domainId):
    return dzclient.create_project(
        domainIdentifier = domainId,
        name = "sample-project"
    )
```

Create an environment profile

You can use the following sample scripts to create an environment profile in Amazon DataZone.

This sample payload is used when the `CreateEnvironmentProfile` API is invoked:

```
Sample Payload
{
    "Content":{
        "project_name": "Admin_project",
        "domain_name": "Drug-Research-and-Development",
        "blueprint_account_region": [
            {
                "blueprint_name": "DefaultDataLake",
                "account_id": ["066535990535",
                "413878397724",
                "676266385322",
                "747721550195",
                "755347404384"
                ],
                "region": ["us-west-2", "us-east-1"]
            },
            {
                "blueprint_name": "DefaultDataWarehouse",
                "account_id": ["066535990535",
                "413878397724",
                "676266385322",
                "747721550195",
                "755347404384"
                ],
                "region":["us-west-2", "us-east-1"]
            }
        ]
    }
}
```

This sample script invokes the `CreateEnvironmentProfile` API:

```
def create_environment_profile(domain_id, project_id, env_blueprints)
        try:
            response = dz.list_environment_blueprints(
                domainIdentifier=domain_id,
                managed=True
            )
            env_blueprints = response.get("items")
            env_blueprints_map = {}
            for i in env_blueprints:
                env_blueprints_map[i["name"]] = i['id']

            print("Environment Blueprint map", env_blueprints_map)
            for i in blueprint_account_region:
                print(i)
                for j in i["account_id"]:
                    for k in i["region"]:
                        print("The env blueprint name is", i['blueprint_name'])
                        dz.create_environment_profile(
                            description='This is a test environment profile created via
 lambda function',
                            domainIdentifier=domain_id,
                            awsAccountId=j,
                            awsAccountRegion=k,

 environmentBlueprintIdentifier=env_blueprints_map.get(i["blueprint_name"]),
                            name=i["blueprint_name"] + j + k + "_profile",
                            projectIdentifier=project_id
                        )
        except Exception as e:
            print("Failed to created Environment Profile")
            raise e
```

This is the sample output payload once the `CreateEnvironmentProfile` API is invoked:

```
{
    "Content":{
        "project_name": "Admin_project",
        "domain_name": "Drug-Research-and-Development",
        "blueprint_account_region": [
            {
                "blueprint_name": "DefaultDataWarehouse",
                "account_id": ["111111111111"],
                "region":["us-west-2"],
                "user_parameters":[
                    {
                        "name": "dataAccessSecretsArn",
                        "value": ""
                    }
                ]
            }
        ]
    }
}
```

Create an environment

You can use the following sample script to create an environment in Amazon DataZone.

```
def create_environment(domain_id, project_id,blueprint_account_region ):
         try:
            #refer to get_domain_id and get_project_id for fetching ids using names.
            sts_client = boto3.client("sts")
            # Get the current account ID
            account_id = sts_client.get_caller_identity()["Account"]
            print("Fetching environment profile ids")
            env_profile_map = get_env_profile_map(domain_id, project_id)

            for i in blueprint_account_region:
                for j in i["account_id"]:
                    for k in i["region"]:
                        print(" env blueprint name", i['blueprint_name'])
                        profile_name = i["blueprint_name"] + j + k + "_profile"
                        env_name = i["blueprint_name"] + j + k + "_env"
                        description = f'This is environment is created for
 {profile_name}, Account {account_id} and region {i["region"]}'
                        try:
                            dz.create_environment(
                                description=description,
                                domainIdentifier=domain_id,

 environmentProfileIdentifier=env_profile_map.get(profile_name),
                                name=env_name,
                                projectIdentifier=project_id
                            )
                            print(f"Environment created - {env_name}")
                        except:
                            dz.create_environment(
                                description=description,
                                domainIdentifier=domain_id,

 environmentProfileIdentifier=env_profile_map.get(profile_name),
                                name=env_name,
                                projectIdentifier=project_id,
                                userParameters= i["user_parameters"]
                            )
                            print(f"Environment created - {env_name}")
```

```
        except Exception as e:
            print("Failed to created Environment")
            raise e
```

Gather metadata from AWS Glue

You can use this sample script to gather metadata from AWS Glue. This script runs on a standard schedule. You can retrieve the parameters from the sample script and make them global. Fetch the project, environment, and domain ID using standard functions. The AWS Glue data source is created and ran at a standard time which can be updated in the cron section of the script.

```
def crcreate_data_source(domain_id, project_id,data_source_name)
        print("Creating Data Source")
        data_source_creation = dz.create_data_source(
            # Define data source : Customize the data source to which you'd like to
 connect
            # define the name of the Data source to create, example: name
 ='TestGlueDataSource'
            name=data_source_name,
            # give a description for the datasource (optional), example:
 description='This is a dorra test for creation on DZ datasources'
            description=data_source_description,
            # insert the domain identifier corresponding to the domain to which the
 datasource will belong, example: domainIdentifier= 'dzd_6f3gst5jjmrrmv'
            domainIdentifier=domain_id,
            # give environment identifier , example: environmentIdentifier=
 '3weyt6hhn8qcvb'
            environmentIdentifier=environment_id,
            # give corresponding project identifier, example: projectIdentifier=
 '6tl4csoyrg16ef',
            projectIdentifier=project_id,
            enableSetting="ENABLED",
            # publishOnImport used to select whether assets are added to the inventory
 and/or discovery catalog .
            # publishOnImport = True : Assets will be added to project's inventory as
 well as published to the discovery catalog
            # publishOnImport = False : Assets will only be added to project's
 inventory.
            # You can later curate the metadata of the assets and choose subscription
 terms to publish them from the inventory to the discovery catalog.
            publishOnImport=False,
            # Automated business name generation : Use AI to automatically generate
 metadata for assets as they are published or updated by this data source run.
            # Automatically generated metadata can be be approved, rejected, or edited
 by data publishers.
            # Automatically generated metadata is badged with a small icon next to the
 corresponding metadata field.
            recommendation={"enableBusinessNameGeneration": True},
            type="GLUE",
            configuration={
                "glueRunConfiguration": {
                    "dataAccessRole": "arn:aws:iam::"
                    + account_id
                    + ":role/service-role/AmazonDataZoneGlueAccess-"
                    + current_region
                    + "-"
                    + domain_id
                    + "",
                    "relationalFilterConfigurations": [
                        {
                            #
                            "databaseName": glue_database_name,
                            "filterExpressions": [
                                {"expression": "*", "type": "INCLUDE"},
                            ],
                            #    "schemaName": "TestSchemaName",
                        },
                    ],
                },
            },
            # Add metadata forms to the data source (OPTIONAL).
            # Metadata forms will be automatically applied to any assets that are
 created by the data source.
            # assetFormsInput=[
            #     {
            #         "content": "string",
            #         "formName": "string",
            #         "typeIdentifier": "string",
            #         "typeRevision": "string",
            #     },
            # ],
            schedule={
                "schedule": "cron(5 20 * * ? *)",
                "timezone": "UTC",
            },
        )
        # This is a suggested syntax to return values
        #        return_values["data_source_creation"] = data_source_creation["items"]
        print("Data Source Created")


//This is the sample response payload after the CreateDataSource API is invoked:

{
    "Content":{
        "project_name": "Admin",
        "domain_name": "Drug-Research-and-Development",
        "env_name": "GlueEnvironment",
        "glue_database_name": "test",
        "data_source_name" : "test",
        "data_source_description" : "This is a test data source"
    }
}
```

Curate and publish a data asset

You can use the following sample scripts to curate and publish data assets in Amazon DataZone.

You can use the following script to create custom form types:

```
def create_form_type(domainId, projectId):
    return dzclient.create_form_type(
        domainIdentifier = domainId,
        name = "customForm",
        model = {
            "smithy": "structure customForm { simple: String }"
        },
        owningProjectIdentifier = projectId,
        status = "ENABLED"
    )
```

You can use the following sample script to create custom asset types:

```
def create_custom_asset_type(domainId, projectId):
    return dzclient.create_asset_type(
        domainIdentifier = domainId,
        name = "userCustomAssetType",
        formsInput = {
            "Model": {
                "typeIdentifier": "customForm",
                "typeRevision": "1",
                "required": False
            }
        },
        owningProjectIdentifier = projectId,
    )
```

You can use the following sample script to create custom assets:

```
def create_custom_asset(domainId, projectId):
    return dzclient.create_asset(
        domainIdentifier = domainId,
        name = 'custom asset',
        description = "custom asset",
        owningProjectIdentifier = projectId,
        typeIdentifier = "userCustomAssetType",
        formsInput = [
            {
                "formName": "UserCustomForm",
                "typeIdentifier": "customForm",
                "content": "{\"simple\":\"sample-catalogId\"}"
            }
        ]
    )
```

You can use the following sample script to create a glossary:

```
def create_glossary(domainId, projectId):
    return dzclient.create_glossary(
        domainIdentifier = domainId,
        name = "test7",
        description = "this is a test glossary",
        owningProjectIdentifier = projectId
    )
```

You can use the following sample script to create a glossary term:

```
def create_glossary_term(domainId, glossaryId):
    return dzclient.create_glossary_term(
        domainIdentifier = domainId,
        name = "soccer",
        shortDescription = "this is a test glossary",
        glossaryIdentifier = glossaryId,
    )
```

You can use the following sample script to create an asset using a system-defined asset type:

```
def create_asset(domainId, projectId):
    return dzclient.create_asset(
        domainIdentifier = domainId,
        name = 'sample asset name',
        description = "this is a glue table asset",
        owningProjectIdentifier = projectId,
        typeIdentifier = "amazon.datazone.GlueTableAssetType",
        formsInput = [
            {
                "formName": "GlueTableForm",
                "content": "{\"catalogId\":\"sample-catalogId\",\"columns\":
[{\"columnDescription\":\"sample-columnDescription\",\"columnName\":\"sample-
columnName\",\"dataType\":\"sample-dataType\",\"lakeFormationTags\":{\"sample-
key1\":\"sample-value1\",\"sample-key2\":\"sample-value2\"}}],\"compressionType\":
\"sample-compressionType\",\"lakeFormationDetails\":{\"lakeFormationManagedTable
\":false,\"lakeFormationTags\":{\"sample-key1\":\"sample-value1\",\"sample-key2\":
\"sample-value2\"}},\"primaryKeys\":[\"sample-Key1\",\"sample-Key2\"],\"region\":
\"us-east-1\",\"sortKeys\":[\"sample-sortKey1\"],\"sourceClassification\":\"sample-
sourceClassification\",\"sourceLocation\":\"sample-sourceLocation\",\"tableArn\":
\"sample-tableArn\",\"tableDescription\":\"sample-tableDescription\",\"tableName\":
\"sample-tableName\"}"
            }
        ]
    )
```

You can use the following sample script to create an asset revision and attach a glossary term:

```
def create_asset_revision(domainId, assetId):
    return dzclient.create_asset_revision(
        domainIdentifier = domainId,
        identifier = assetId,
        name = 'glue table asset 7',
        description = "glue table asset description update",
        formsInput = [
            {
                "formName": "GlueTableForm",
                "content": "{\"catalogId\":\"sample-catalogId\",\"columns\":
[{\"columnDescription\":\"sample-columnDescription\",\"columnName\":\"sample-
columnName\",\"dataType\":\"sample-dataType\",\"lakeFormationTags\":{\"sample-
key1\":\"sample-value1\",\"sample-key2\":\"sample-value2\"}}],\"compressionType\":
\"sample-compressionType\",\"lakeFormationDetails\":{\"lakeFormationManagedTable
\":false,\"lakeFormationTags\":{\"sample-key1\":\"sample-value1\",\"sample-key2\":
\"sample-value2\"}},\"primaryKeys\":[\"sample-Key1\",\"sample-Key2\"],\"region\":
\"us-east-1\",\"sortKeys\":[\"sample-sortKey1\"],\"sourceClassification\":\"sample-
sourceClassification\",\"sourceLocation\":\"sample-sourceLocation\",\"tableArn\":
\"sample-tableArn\",\"tableDescription\":\"sample-tableDescription\",\"tableName\":
\"sample-tableName\"}"
            }
        ],
        glossaryTerms = ["<glossaryTermId:>"]
    )
```

You can use the following sample script to publish an asset:

```
def publish_asset(domainId, assetId):
    return dzclient.create_listing_change_set(
        domainIdentifier = domainId,
        entityIdentifier = assetId,
        entityType = "ASSET",
        action = "PUBLISH",
    )
```

Search the data catalog and subscribe to data

You can use the following sample scripts to search the data catalog and subscribe to data:

```
def search_asset(domainId, projectId, text):
    return dzclient.search(
        domainIdentifier = domainId,
        owningProjectIdentifier = projectId,
        searchScope = "ASSET",
        searchText = text,
    )
```

You can use the following sample script to get the listing ID for the asset:

```
def search_listings(domainId, assetName, assetId):
    listings = dzclient.search_listings(
        domainIdentifier=domainId,
        searchText=assetName,
        additionalAttributes=["FORMS"]
    )

    assetListing = None
    for listing in listings['items']:
        if listing['assetListing']['entityId'] == assetId:
            assetListing = listing

    return listing['assetListing']['listingId']
```

You can use the following sample scripts to create a subscription request using the listing ID:

```
create_subscription_response = def create_subscription_request(domainId, projectId,
 listingId):
    return dzclient.create_subscription_request(
        subscribedPrincipals=[{
            "project": {
                "identifier": projectId
            }
        }],
        subscribedListings=[{
            "identifier": listingId
        }],
        requestReason="Give request reason here."
    )
```

Using the `create_subscription_response` above, get the `subscription_request_id`, and then accept/approve the subscription using the following sample script:

```
subscription_request_id = create_subscription_response["id"]

def accept_subscription_request(domainId, subscriptionRequestId):
    return dzclient.accept_subscription_request(
        domainIdentifier=domainId,
        identifier=subscriptionRequestId
    )
```

Other useful sample scripts

You can use the following sample scripts to complete various tasks as you work with your data in Amazon DataZone.

Use the following sample script to list existing Amazon DataZone domains:

```
def list_domains():
    datazone = boto3.client('datazone')
    response = datazone.list_domains(status='AVAILABLE')
    [print("%12s | %16s | %12s | %52s" % (item['id'], item['name'],
 item['managedAccountId'], item['portalUrl'])) for item in response['items']]
    return
```

Use the following sample script to list existing Amazon DataZone projects:

```
def list_projects(domain_id):
    datazone = boto3.client('datazone')
    response = datazone.list_projects(domainIdentifier=domain_id)
    [print("%12s | %16s " % (item['id'], item['name'])) for item in response['items']]
    return
```

Use the following sample script to list existing Amazon DataZone metadata forms:

```
def list_metadata_forms(domain_id):
    datazone = boto3.client('datazone')
    response = datazone.search_types(domainIdentifier=domain_id,
        managed=False,
        searchScope='FORM_TYPE')
    [print("%16s | %16s | %3s | %8s" % (item['formTypeItem']['name'],
 item['formTypeItem']['owningProjectId'],item['formTypeItem']['revision'],
 item['formTypeItem']['status'])) for item in response['items']]
    return
```

Managing Amazon DataZone domains and user access

Topics

Create domains

> **Note**
>
> If you are using Amazon DataZone with AWS Identity Center to provide access to SSO users and groups, then currently your Amazon DataZone domain must be in the same AWS Region as your AWS Identity Center instance.

Amazon DataZone, a domain is an organizing entity for connecting together your assets, users, and their projects. For more information, see Amazon DataZone terminology and concepts.

To create an Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions necessary to create a domain.

Additional IAM roles are needed by Amazon DataZone to perform actions on behalf of domain users with a default configuration. You can create these IAM roles in advance, or have Amazon DataZone create them for you. If you want Amazon DataZone to create these IAM roles for you during the domain creation process, then for domain creation you must assume an IAM role with role creation permissions. See Create a custom policy for IAM permissions to enable the Amazon DataZone service console simplified role creation . Depending on your domain creation choices, Amazon DataZone will create up to four new IAM roles for you: **AmazonDataZoneDomainExecutionRole**, **AmazonDataZoneGlueManageAccessRole**, **AmazonDataZoneRedshiftManageAccessRole**, and **AmazonDataZoneProvisioningRole**.

Complete the following procedure to create an Amazon DataZone domain.

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and use the region selector in the top navigation bar to choose the appropriate AWS Region.
2. Choose **Create domain** and provide values for the following fields:

   - **Name** - specify a friendly name for the domain. Once the domain is created this name cannot be changed.
   - **Description** - (optional) specify a domain description.
   - **Data encryption** - your Amazon DataZone domain, metadata, and reporting data is encrypted by the AWS Key Management Service (KMS) using a key specific to your Amazon DataZone. Use this field to specify whether you want to use an AWS owned key or choose a different AWS KMS key.

     For more information about using customer managed keys, see Data encryption at rest for Amazon DataZone. If you use your own KMS key for data encryption, you must include the following statement in your default AmazonDataZoneDomainExecutionRole.

     ```
     {
       "Version": "2012-10-17",
       "Statement": [
         {
           "Sid": "Statement1",
           "Effect": "Allow",
           "Action": [
             "kms:Decrypt",
             "kms:DescribeKey",
             "kms:GenerateDataKey"
           ],
           "Resource": [
             "*"
           ]
         }
       ]
     }
     ```

- **Service access** - choose whether to have Amazon DataZone create and use a new **DomainExecutionRole** for you, or choose an existing IAM role.
- **Quick setup** - (optional) check this box to get started faster by having Amazon DataZone set-up your account for data consumption and publishing. Amazon DataZone will create three IAM roles for provisioning, ingesting, and managing access to AWS Glue and Amazon Redshift resources, create a new Amazon S3 bucket, create an administrative Amazon DataZone project, and create environment profiles for the data lake and data warehouse default blueprints.
- Tags - (optional) specify AWS tags (key and value pairs) for the domain.
- Once the domain is successfully created, your browser should be refreshed to display your new Amazon DataZone domain's details page.

Edit domains

In Amazon DataZone, a domain is an organizing entity for connecting together your assets, users, and their projects. For more information, see Amazon DataZone terminology and concepts.

After you create an Amazon DataZone domain, you can later edit the domain to: change the description, enable IAM Identity Center, and add, edit, or remove tag keys and their values. To edit an Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions necessary to edit a domain.

To edit a domain, complete the following steps:

1. Sign in to the AWS Management Console and open the Amazon DataZone console at https://console.aws.amazon.com/datazone.
2. Choose **View domains** and choose the domain's name from the list. The name is a hyperlink.
3. On the details page for the domain, choose **Edit**.
4. - Edit the **Description**.
   - Set the **IAM Identity Center settings**. Learn more about these settings in Setting up AWS IAM Identity Center for Amazon DataZone.
   - Add, edit, or remove **Tag** keys and their values.
5. Once you've made your edits, choose **Update domain**.

Delete domains

In Amazon DataZone, a domain is an organizing entity for connecting together your assets, users, and their projects. For more information, see Amazon DataZone terminology and concepts.

The act of deleting a domain is final. Deletion irrevocably removes every Amazon DataZone entity, including data sources, projects, environments, assets, glossaries, and metadata forms. Deletion does not delete non-Amazon DataZone AWS resources that Amazon DataZone may have helped you create, such as IAM roles, S3 buckets, AWS Glue databases, and subscription grants via LakeFormation or Redshift. If you no longer need these resources, delete them in the respective AWS service.

To prevent someone from deleting a domain maliciously, deleting a domain requires administrative IAM permissions for Amazon DataZone, which you can configure with IAM. To prevent someone from deleting a domain accidentally, deleting a domain requires a confirmation word (in the Amazon DataZone console).

To delete a domain, complete the following steps:

1. Sign in to the AWS Management Console and open the Amazon DataZone console at https://console.aws.amazon.com/datazone.
2. Choose **View domains** and choose the domain’s name from the list. The name is a hyperlink.
3. Choose **Delete** and review the informational warnings.
4. Type in the requested text to confirm that you understand these warnings. Choose **Delete**.

> ⚠ **Important**
>
> Deleting your domain is an irrevocable action that cannot be undone by you or by AWS.

> ⓘ **Note**
>
> When you or your domain users create an environment in a project, Amazon DataZone creates AWS resources in your domain or associated accounts to provide you and your domain users with functionality. Below is the list of AWS resources that Amazon DataZone may create for projects in your domain, along with the default name. Deleting a domain does not delete any of these AWS resources in your AWS accounts.

- IAM roles: datazone_usr_<environmentId>.
- Glue databases: (1) <environmentName>_pub_db-*, (2) <environmentName>_sub_db-*. If there was already an existing database of this name, Amazon DataZone will add the environment ID.
- Athena workgroups: <environmentName>-*. If there was already an existing workgroup of this name, Amazon DataZone will add the environment ID.
- CloudWatch log group: datazone_<environmentId>

Enable IAM Identity Center for Amazon DataZone

Note

To complete this procedure, you must have AWS IAM Identity Center enabled in the same AWS Region as your Amazon DataZone domain.

You can provide SSO users and groups with access to your Amazon DataZone data portal using AWS IAM Identity Center. After completing Setting up AWS IAM Identity Center for Amazon DataZone, you can enable your SSO users and groups to access your Amazon DataZone domain data portal.

To enable AWS IAM Identity Center for use with your Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console and Create a custom policy for IAM permissions to enable the Amazon DataZone service console simplified role creation to obtain the minimum permissions necessary to enable IAM Identity Center for use with Amazon DataZone.

Complete the following procedure to enable the AWS IAM Identity Center for Amazon DataZone.

1. Sign in to the AWS Management Console and open the DataZone console at https://console.aws.amazon.com/datazone.
2. Select **View domains** and choose the domain's name from the list. The name is a hyperlink.
3. On the detail page for the domain, choose **Edit**.

   - Select the checkbox for **Enable users in IAM Identity Center**.

- Choose between the two user assignment modes. Once your domain is updated with your selection, it cannot be changed later.
  - With **Implicit user assignment**, any user added to your IAM Identity Center directory can access your Amazon DataZone domain.
  - With **Explicit user assignment**, you will add specific users or groups from you IAM Identity Center directory to provide them access to your Amazon DataZone domain. You will add and remove these users and groups later in the Amazon DataZone Console.

4. Once you are satisfied with your selection, choose **Update domain**.

Disable IAM Identity Center for Amazon DataZone

Disabling AWS IAM Identity Center for an Amazon DataZone domain will remove access for all SSO users.

> **Note**
>
> Disabling IAM Identity Center will not stop billing for SSO users. To stop billing for SSO users, you must deactivate them in your domain. Billing continues until the end of the month in which a user is deactivated. To deactivate users, see Manage users in the Amazon DataZone console.

You can provide SSO users and groups with access to your Amazon DataZone data portal using AWS IAM Identity Center. If you have enabled AWS IAM Identity Center for Amazon DataZone, you can later disable access for all users.

To disable AWS IAM Identity Center for use with your Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console and Create a custom policy for IAM permissions to enable the Amazon DataZone service console simplified role creation to obtain the minimum permissions necessary to disable IAM Identity Center from use with Amazon DataZone.

Complete the following procedure to disable the AWS IAM Identity Center for Amazon DataZone.

1. Sign in to the AWS Management Console and open the DataZone console at https://console.aws.amazon.com/datazone.
2. Select **View domains** and choose the domain's name from the list. The name is a hyperlink.

3. Copy the **Amazon Resource Name (ARN)** for your domain, which starts with arn:aws:datazone:<regionName>:<accountId>:domain/<domainName>.
4. Open the IAM Identity Center console at https://console.aws.amazon.com/singlesignon/.
5. Choose **Applications**.
6. Choose the domain for which you want to disable AWS IAM Identity Center, which as a result will remove access to the domain's data portal for all SSO users. You can use the **Filter** menu and the search box to filter the list of applications.
7. From the **Actions** menu, choose **Disable**.
8. SSO users will lose access to the Amazon DataZone domain.
9. To re-enable AWS IAM Identity Center for the Amazon DataZone domain, choose the domain for which you want to re-enable AWS IAM Identity Center, and from the **Actions** menu, choose **Enable**.

Manage users in the Amazon DataZone console

Your users can access the Amazon DataZone data portal by using either their AWS credentials or single sign-on (SSO) credentials. To manage users in the Amazon DataZone console for an Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions necessary to manage users in the Amazon DataZone console.

Topics

- Manage IAM roles and users
- Manage SSO users
- Manage SSO groups

Manage IAM roles and users

IAM roles and users are created using AWS Identity and Access Management (IAM) and gain access to your Amazon DataZone domains through permissions attached to them via policies. For more information, see Configure the IAM permissions required to use the Amazon DataZone data portal. In the current release of Amazon DataZone, an administrator from an Amazon DataZone domain owner account, can create IAM user profiles for users in their own account or for users in the associated accounts. An administrator from an Amazon DataZone domain owner account can also set an existing user's status to Assigned or Unassigned (as in assigned or unassigned to use Amazon DataZone) or activate or deactivate any existing user.

1. Sign in to the AWS Management Console and open the DataZone console at https://console.aws.amazon.com/datazone.
2. Select **View domains** and choose the domain's name from the list. The name is a hyperlink.
3. On the details page for the domain, choose **User management**.
4. To add a user IAM user in the Amazon DataZone domain owner account or in the associated account, choose **Add** and then choose **Add IAM users**.
5. On the **Add users** page, choose **Current account** or **Associated account**, use the **Find and add users or roles** field to find the users that you want to add, and then choose **Add users**.
6. To view an existing IAM user's status, on the **User management** page, choose **IAM Users** in the user type drop-down menu.

    - The **Name** column shows the ARN of the IAM user or role.
    - The **Status** column shows the current status of the IAM user or role in the domain.
        - Assigned means that the IAM user has been assigned to use Amazon DataZone.
        - Unassigned means that the IAM user has been unassigned to use Amazon DataZone.
        - Activated means that the IAM user or role has called an API, issued a command (via Command Line Interface), or accessed the Amazon DataZone portal for your domain, and you are being billed for the user's subscription.
        - Deactivated means that the IAM user or role has their access blocked to your Amazon DataZone domain.
7. To deactivate an IAM user or role that is currently activated, check the box next to the user and select **Deactivate** from the **Actions** menu. The user will lose access to the Amazon DataZone domain. Billing for the user will end at the end of the current calendar month.
8. To activate an IAM user or role that is currently deactivated, check the box next to the user and select **Activate** from the **Actions** menu. The user will gain access to the Amazon DataZone domain if the IAM user or role has appropriate permissions. Billing for the user will start again.

Manage SSO users

SSO users are created or synchronized with your identity provider in AWS IAM Identity Center. For more information, see Setting up AWS IAM Identity Center for Amazon DataZone and Enable

IAM Identity Center for Amazon DataZone to enable and configure AWS IAM Identity Center for Amazon DataZone. You can view the list of SSO users assigned to the domain, add SSO users, and remove SSO users.

1. Sign in to the AWS Management Console and open the DataZone console at https://console.aws.amazon.com/datazone.
2. Select **View domains** and choose the domain's name from the list. The name is a hyperlink.
3. On the details page for the domain, scroll down and choose **User management**.
4. For user type, select **SSO Users** to view the current list of SSO users.

   - The **Name** column shows the SSO user's name.
   - The **Status** column shows the current status of the SSO user in the domain.
     - Assigned means that the SSO user has been explicitly assigned to the domain. As a result, the user has access to Amazon DataZone. This status is only used when your domain's identity provider mode is set to explicit assignment.
     - Activated means that the SSO user has accessed the Amazon DataZone portal for the domain and you are being billed for the user's subscription. Activation happens automatically.
     - Deactivated means that the SSO user's access is blocked to the domain's data portal. Billing for the user ended at the end of the month in which their access was deactivated.
     - Removed means that the SSO user was previously assigned to the domain, but removed before they ever accessed.
5. Add SSO users by choosing **Add** and **Add users**. This option is unavailable if the domain is set to implicit user assignment, which means that all users in the identity pool have access to the Amazon DataZone domain.

   - On the **Add users** page, search for the aliases of the users you want to add. A list will appear below the search box with potential matches.
   - Choose the user you want to add. Their alias will appear as a chip below the search box.
   - When you are satisfied with the list of users you want to add, choose **Add user(s)**.
   - The users are assigned to the Amazon DataZone domain with a status of **Assigned**.
   - When the user first accessed the domain's data portal, the status will change automatically to **Activated**, and you will start being billed for the user's subscription.
6. Remove an **Assigned** SSO user by selecting the user and choosing **Disable** from the **Actions** menu. As a result, the user will lose access to the Amazon DataZone domain. The user's

status will show as **Removed**. This option is unavailable if the domain is set to implicit user assignment.

7. Deactivate an **Activated** SSO user by selecting the user and choosing **Deactivate** from the **Actions** menu. As a result, the user's access to the Amazon DataZone domain will be lost and blocked. Billing will continue for the user's subscription until the end of the month. The user's status will show as **Deactivated**.
8. Activate a **Deactivated** SSO user by selecting the user and choosing **Activate** from the **Actions** menu. As a result, the user will regain access to the Amazon DataZone domain. Billing will begin immediately. The user's will show as **Activated**.

Manage SSO groups

SSO groups are created or synchronized with your identity provider in AWS IAM Identity Center. For more information, see Setting up AWS IAM Identity Center for Amazon DataZone and Enable IAM Identity Center for Amazon DataZone to enable and configure AWS IAM Identity Center for Amazon DataZone. You can view the list of SSO groups assigned to the domain, add SSO groups, and remove SSO groups.

1. Sign in to the AWS Management Console and open the DataZone console at https://console.aws.amazon.com/datazone.
2. Select **View domains** and choose the domain's name from the list. The name is a hyperlink.
3. On the details page for the domain, scroll down and choose **User management**.
4. For user type, select **SSO Groups** to view the current list of SSO groups.
   - The **Name** column shows the SSO group's name.
   - The **Status** column shows the current status of the SSO group in the domain.
     - **Assigned** means that the SSO group has been explicitly assigned to the domain. As a result, all users in the group have access to the domain's data portal (unless the user is deactivated).
     - **Not Assigned** means that the SSO group has been removed from the domain. Users in the group do not have access to the domain's data portal via their membership in this group.
5. Add SSO groups by choosing **Add** and **Add groups**. This option is unavailable if the domain is set to implicit user assignment, which means that all users in the identity pool have access to the Amazon DataZone domain regardless of group membership.

- On the **Add groups** page, search for the aliases of the groups you want to add. A list will appear below the search box with potential matches.
- Choose the group you want to add. Their alias will appear as a chip below the search box.
- When you are satisfied with the list of groups you want to add, choose **Add group(s)**.
- The groups are assigned to the Amazon DataZone domain with a status of **Assigned**.
- When a member of the group accesses the domain's data portal, the status will change automatically to **Activated**, and you will start being billed for the user’s subscription.

6. Remove an **Assigned SSO group** by selecting the group and choosing **Unassign** from the **Actions** menu. As a result, the group will lose access to the Amazon DataZone domain. The group’s status will show as **Not Assigned**. Users that gained their access to Amazon DataZone via their membership in this group will lose access. This option is unavailable if the domain is set to implicit user assignment. To stop billing for users whose access is removed by unassigning their group, you will need to next manually select and **Deactivate** their user profiles.

Managing user permissions in the Amazon DataZone data portal

In the current release of Amazon DataZone, the default authorization mechanism enables all authenticated users (IAM and SSO) of the Amazon DataZone domains to create projects, create entities within the projects, and conduct searches. Project members must still abide by the permissions given to them per their designated project owner or project contributor roles.

Working with the Amazon DataZone built-in blueprints

A blueprint with which an environment is created defines what tools and services members of the project to which the environment belongs can use as they work with assets in the Amazon DataZone catalog. In the current release of Amazon DataZone, there are the following built-in blueprints:

- Data lake blueprint
- Data warehouse blueprint
- Amazon SageMaker blueprint

Topics

Enable built-in blueprints in the AWS account that owns the Amazon DataZone domain

A blueprint with which an environment is created defines what tools and services members of the project to which the environment belongs can use as they work with assets in the Amazon DataZone catalog.

In the current release of Amazon DataZone, there are several built-in blueprints: data lake blueprint, data warehouse blueprint, and Amazon SageMaker blueprint.

- Data lake blueprint contains the definition for launching and configuring a set of services (AWS Glue, AWS Lake Formation, Amazon Athena) to publish and use data lake assets in the Amazon DataZone catalog.
- Data warehouse blueprint contains the definition for launching and configuring a set of services (Amazon Redshift) to publish and use Amazon Redshift assets in the Amazon DataZone catalog.
- Amazon SageMaker blueprint contains the definition for launching and configuring a set of services (Amazon SageMaker Studio) to publish and use Amazon SageMaker assets in the Amazon DataZone catalog.

For more information, see Amazon DataZone terminology and concepts.

While creating an Amazon DataZone domain, you have the option to choose the **Quick setup** which automatically enables the default data lake and the default data warehouse built-in blueprints as part of the domain creation process. **Quick setup** also creates default environment profiles and default environments for you using these built-in blueprints.

If you don't choose **Quick setup** as part of creating your Amazon DataZone domain, you can use the procedure below to enable the available built-in blueprints in the AWS account that houses this Amazon DataZone domain. You must enable these built-in blueprints before you can use them to create envrionment profiles and environments in this domain.

To enable built-in blueprints in an Amazon DataZone domain via the Amazon DataZone management console, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions.

Enable built-in blueprints in an Amazon DataZone domain

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Choose **View domains** and choose the domain where you want to enable one or more built-in blueprints.
3. On the domain details page, navigate to the **Blueprints** tab.
4. From the **Blueprints** list, choose either the **DefaultDataLake** or the **DefaultDataWarehouse**, or the **Amazon SageMaker** blueprint.
5. On the chosen blueprint's details page, choose **Enable in this account**.
6. On the Permissions and resources page, specify the following:

   - If you're enabling the **DefaultDataLake** blueprint, for **Glue Manage Access role**, specify a new or existing service role that grants Amazon DataZone authorization to ingest and manage access to tables in AWS Glue and AWS Lake Formation.
   - If you're enabling the **DefaultDataWarehouse** blueprint, for **Redshift Manage Access role**, specify a new or existing service role that grants Amazon DataZone authorization to ingest and manage access to datashares, tables and views in Amazon Redshift.
   - If you're enabling the **Amazon SageMaker** blueprint, for **SageMaker Manage Access role**, specify a new or existing service role that grants Amazon DataZone permissions to publish

Amazon SageMaker data to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to Amazon SageMaker published assets in the catalog.

> ⚠ **Important**
>
> When you're enabling the **Amazon SageMaker** blueprint, Amazon DataZone checks whether the following IAM roles for Amazon DataZone exist in the current account and region. If these roles do not exist, Amazon DataZone automatically creates them.
>
> - AmazonDataZoneGlueAccess-<region>-<domainId>
> - AmazonDataZoneRedshiftAccess-<region>-<domainId>

- For **Provisioning role**, specify a new or existing service role that grants Amazon DataZone authorization to create and configure environment resources using AWS CloudFormation in the environment account and region.
- If you're enabling the **Amazon SageMaker** blueprint, for the **Amazon S3 bucket for SageMaker-Glue data source**, specify an Amazon S3 bucket that is to be used by all SageMaker environments in the AWS account. The bucket prefix that you specify must be one of the following:
  - amazon-datazone*
  - datazone-sagemaker*
  - sagemaker-datazone*
  - DataZone-Sagemaker*
  - Sagemaker-DataZone*
  - DataZone-SageMaker*
  - SageMaker-DataZone*

7. Choose **Enable blueprint**.

Once you enable the chosen blueprint(s), you can control which projects can use the blueprint(s) in your account to create environment profiles. You can do this by assigning managing projects to the blueprint’s configuration.

Specify managing projects on enabled blueprints

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.

2. Choose **View Domains** and then choose the domain where you want to add the managing project(s) for the chosen blueprint(s).
3. Choose the **Blueprints** tab and then choose the blueprint that you want to work with.
4. By default, all projects within the domain can use the DefaultDataLake or DefaultDataWareshouse, or the Amazon SageMaker blueprints in the account to create environment profiles. However, you can restrict this by assigning managing projects to the blueprints. To add managing projects, choose **Select managing project**, then choose the projects that you want to add as managing projects from the drop down menu, and then choose **Select managing projects(s)**.

Once you enable the DefaultDataWarehouse blueprint in your AWS account, you can add parameter sets to the blueprint configuration. A parameter set is a group of keys and values, required for Amazon DataZone to establish a connection to your Amazon Redshift cluster and is used to create data warehouse environments. These parameters include the name of your Amazon Redshift cluster, database, and the AWS secret that holds credentials to the cluster.

Adding parameter sets to the DefaultDataWarehouse blueprint

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Choose **View domains** and then choose the domain where you want to add the parameter set.
3. Choose the **Blueprints** tab and then choose the DefaultDataWareshouse blueprint to open the blueprint details page.
4. Under the **Parameter sets** tab on the blueprint details page, choose **Create parameter set**.

   - Provide a Name for the parameter set.
   - Optionally, provide a description for the parameter set.
   - Select a region
   - Select either Amazon Redshift cluster or Amazon Redshift Serverless.
   - Select the AWS secret ARN that holds the credentials to the selected Amazon Redshift cluster or the Amazon Redshift Serverless workgroup. The AWS secret must be tagged with the `AmazonDataZoneDomain : [Domain_ID]` tag in order to be eligible for use within a parameter set.
     - If you do not have an existing AWS secret, you can also create a new secret by choosing **Create New AWS Secret**. This opens a dialog box where you can provide the name of

the secret, username, and password. Once you choose **Create New AWS Secret**, Amazon DataZone creates a new secret in the AWS Secrets Manager service and ensures that the secret is tagged with the domain in which you are trying to create the parameter set.

- If you chose Amazon Redshift cluster in the step above, now choose a cluster from the dropdown. If you chose Amazon Redshift workgroup in the step above, now choose a workgroup from the drop down.
- Enter the name of the database within the selected Amazon Redshift cluster or Amazon Redshift Serverless workgroup.
- Choose **Create parameter set**.

Once you enable the Amazon SageMaker blueprint in your AWS account, you can add parameter sets to the blueprint configuration. A parameter set is a group of keys and values, required for Amazon DataZone to establish a connection to your Amazon SageMaker and is used to create sagemaker environments.

Adding parameter sets to the Amazon SageMaker blueprint

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Choose **View domains** and then choose the domain that contains the enabled blueprint where you want to add the parameter set.
3. Choose the **Blueprints** tab and then choose the Amazon SageMaker blueprint to open the blueprint's details page.
4. Under the **Parameter sets** tab on the blueprint details page, choose **Create parameter set**, and then specify the following:
   - Provide a **Name** for the parameter set.
   - Optionally, provide a **Description** for the parameter set.
   - Specify the Amazon SageMaker domain authentication type. You can choose either IAM or IAM Identity Center (SSO).
   - Specify an AWS region.
   - Specify an AWS KMS key for data encryption. You can choose an existing key or create a new key.
   - Under **Environment parameters**, specify the following:

    - VPC ID - the ID that you're using for the VPC of the Amazon SageMaker environment. You can specify an existing or create a new VPC.
    - Subnets - one or more IDs for a range of IP addresses for specific resources within your VPC.
    - Network access - choose either **VPC only** or **Public internet only**.
    - Security group - the security group to use when configuring VPC and subnets.
- Under Data source parameters, choose one of the following:
    - AWS Glue only
    - AWS Glue + Amazon Redshift Serverless. If you choose this option, specify the following:
        - Specify the AWS secret ARN that holds the credentials to the selected Amazon Redshift cluster. The AWS secret must be tagged with the `AmazonDataZoneDomain : [Domain_ID]` tag in order to be eligible for use within a parameter set.

          If you do not have an existing AWS secret, you can also create a new secret by choosing **Create New AWS Secret**. This opens a dialog box where you can provide the name of the secret, username, and password. Once you choose **Create New AWS Secret**, Amazon DataZone creates a new secret in the AWS Secrets Manager service and ensures that the secret is tagged with the domain in which you are trying to create the parameter set.
        - Specify the Amazon Redshift workgroup you want to use when creating environments.
        - Specify the name of the database (within the workgroup you've chosen) that you want to use when creating environments.
    - AWS Glue only + Amazon Redshift Cluster
        - Specify the AWS secret ARN that holds the credentials to the selected Amazon Redshift cluster. The AWS secret must be tagged with the `AmazonDataZoneDomain : [Domain_ID]` tag in order to be eligible for use within a parameter set.

          If you do not have an existing AWS secret, you can also create a new secret by choosing **Create New AWS Secret**. This opens a dialog box where you can provide the name of the secret, username, and password. Once you choose **Create New AWS Secret**, Amazon DataZone creates a new secret in the AWS Secrets Manager service and ensures that the secret is tagged with the domain in which you are trying to create the parameter set.
        - Specify the Amazon Redshift cluster you want to use when creating environments.
        - Specify the name of the database (within the cluster you've chosen) that you want to use when creating environments.

5. Choose **Create parameter set**.

Add Amazon SageMaker as a trusted service in the AWS account that owns the Amazon DataZone domain

If you've enabled the Amazon SageMaker blueprint, you must also add SageMaker as one of the trusted services within Amazon DataZone. To do this, complete the following procedure:

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Choose **View domains** and then choose the domain that contains the enabled SageMaker blueprint.
3. Choose the **Trusted services**, then choose the **Amazon SageMaker**, and then choose **Enable**.

Working with custom AWS service blueprints in Amazon DataZone

In Amazon DataZone, custom AWS service blueprints enable you optimize resource usage and costs by configuring Amazon DataZone to use your own existing AWS Identity and Access Management (IAM) roles and AWS services that you already have set up in your organization.

A blueprint with which a Amazon DataZone environment is created defines what tools and services members of the project to which the environment belongs can use as they work with assets in the Amazon DataZone catalog. In the current release of Amazon DataZone, there are the following built-in blueprints:

- Data lake blueprint
- Data warehouse blueprint
- Amazon SageMaker blueprint

With Amazon DataZone custom AWS service blueprints, you can create environments and projects that are customized to any AWS services that you are currently using in your organization. With custom blueprints, you can include Amazon DataZone in your existing data pipelines by configuring it to use your existing IAM roles to enhance governance across infrastructure setup and collaborate on business initiatives.

Topics

Enable a custom AWS service blueprint

Complete the following procedure to enable a custom AWS service blueprint in your domain.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose **View domains** and choose the domain in which you want to enable a custom AWS service blueprint.
3. Choose the **Blueprints** tab, then choose the **AWS service** bluepint from the list of available blueprints, and then choose **Enable**.

Create an environment using a custom AWS service blueprint

Complete the following procedure to create an environment using a custom AWS service blueprint.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose **View domains** and choose the domain in which your custom AWS service blueprint is enabled.
3. Choose the **Blueprints** tab, then choose the enabled **AWS service** bluepint, and then choose **Create environment**.
4. On the **Create environment** page, specify the following and then choose **Create environment**:

   - **Name** - specify the name for the environment.
   - **Description** - specify the description for the environment.
   - **Project** - specify a new or existing owning project for the environment. Projects enable groups of users to discover, publish, subscribe to, and consume assets in Amazon DataZone. This environment will be available to all the members of the specified project. All environments are owned by projects whose users have access to the environment.
   - **Environment role** - specify an existing IAM role that will grant Amazon DataZone access to your existing AWS services and resources, such as Amazon S3 and AWS Glue, in this environment.

   > **Note**
   >
   > Amazon DataZone does not provision this role for you. You must have an existing IAM role with permissions to your existing AWS services and resources that you want to enable in this environment.

> Make sure that this IAM role has the minimum required permissions, in other words, is scoped down to provide access only to the AWS services and resources that you want to enable in this environment.

- **AWS region** - specify an AWS region in which you want to create this environment.

Create actions in a custom AWS service environment

Complete the following procedure to create actions in a custom AWS service environment. By creating actions in a custom AWS service environment, you're adding deep links to the Amazon DataZone data portal to the analytics tool that are available in this environment.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose **View domains** and choose the domain in which your custom AWS service blueprint is enabled.
3. Choose the **Blueprints** tab, then choose the enabled **AWS service** bluepint, and then choose the AWS service environment where you want to add actions.
4. On the AWS console links page, choose links (actions) from the **Popular AWS links** or the **Custom AWS links** sections to enable deep links to your Amazon S3 buckets, Amazon Athena workgroups, AWS Glue jobs, or to any other custom AWS console resource from this environment via the Amazon DataZone data portal.
5. If you navigate to this environment in the data portal by using the **Data portal link** from the **Summary** section of this environment, you can see the deep links that you've added under the **Analytics tools** section.

Add project members to a custom AWS service environment

Complete the following procedure to add project members to an AWS service environment.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose the **Projects** tab and then choose the project within a AWS service environment to which you want to add members.

3. Choose **Add** and then, on the **Add members** page, find and add members from **IAM users**, **SSO users**, or **SSO groups**. Specify an assigned project role of either an **Owner** or a **Contributor**. When you're finished finding and adding members, choose **Add members**.

Configure a data source in an AWS service environment

Complete the following procedure to configure a data source in an AWS service environment.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose the **Blueprints** tab and then choose the custom AWS service blueprint.
3. Under **Created environments**, choose the AWS service environment where you want to configure a data source.
4. Choose the **Data sources** tab, choose **Add**, specify the following, and then choose **Add**.

   - **Name** - data source name.
   - **Resource** - choose either AWS Glue or Amazon Redshift.
     - For AWS Glue, specify the resource database.
     - For Amazon Redshift, choose either **Cluster** or **Serverless**, and then specify the **Redshift Credentials**, including a new or existing AWS secret, a cluster or serverless workgroup you want to use when creating environments, the database you want to use when creating environments, and the schema within the specified database.
   - **Permissions** - specify a manage access role that will provide Amazon DataZone with authorization to ingest and manage access to tables in AWS Lake Formation (for AWS Glue) or that will provide Amazon DataZone with authorization to ingest and manage access to tables in Amazon Redshift.
   - **Use for data consumption** - in Amazon DataZone, project members can consume data through subscription targets which Amazon DataZone uses to enable the access to the data for which you have subscribed in your projects. Specify whether to also add this data source as a subscription target.

Configure a subscription target in an AWS service environment

Complete the following procedure to configure a subscription target in an AWS service environment.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose the **Blueprints** tab and then choose the AWS service blueprint.
3. Under **Created environments**, choose the AWS service environment where you want to configure a subscription target.
4. Choose the **Subscription targets** tab, choose **Add**, specify the following, and then choose **Add**.

    - **Name** - subscription target name.
    - **Resource** - choose either AWS Glue or Amazon Redshift.
        - For AWS Glue, specify the resource database.
        - For Amazon Redshift, choose either **Cluster** or **Serverless**, and then specify the **Redshift Credentials**, including a new or existing AWS secret, a cluster or serverless workgroup you want to use when creating environments, the database you want to use when creating environments, and the schema within the specified database.
    - **Permissions** - specify a manage access role that will provide Amazon DataZone with authorization to ingest and manage access to tables in AWS Lake Formation (for AWS Glue) or that will provide Amazon DataZone with authorization to ingest and manage access to tables in Amazon Redshift.
    - **Use for data consumption** - in Amazon DataZone, you can publish data to the data catalog through a data source that allows for metadata ingestion. Specify whether to also add this subscription target as a data source.

Working with associated accounts to publish and consume data

Associating your AWS accounts with your Amazon DataZone domain enables domain users to publish and consume data from these AWS accounts. There are three steps to setting up an account association.

- First, share the domain with the desired AWS account by requesting association. Amazon DataZone uses AWS Resource Access Manager (RAM) if the AWS account is different from the domain's AWS account. An account association can only be initiated by the Amazon DataZone domain.
- Second, have the account owner accept the association request.
- Third, have the account owner enable the desired environment blueprints. By enabling a blueprint, the account owner is providing users in the domain the IAM roles and resource configurations necessary to create and access resources in their account, such as AWS Glue databases and Amazon Redshift clusters.

Topics

Request association with other AWS accounts

> **Note**
>
> By sending an association request to another AWS account, you are sharing your domain with the other AWS account with AWS Resource Access Manager (RAM). Be sure to check the accuracy of the account ID that you enter.

To request association with other AWS accounts in the Amazon DataZone console for an Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions necessary to request an account association.

Complete the following procedure to request association with other AWS accounts.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose **View domains** and choose the domain's name from the list. The name is a hyperlink.
3. Scroll down to the **Associated accounts** tab and choose **Request association**.
4. Enter the IDs of the accounts that you want to request association. When you are satisfied with the list of account IDs, choose **Request association**.
5. Under RAM Policy, specify the RAM policy for account association. You can either choose `AWSRAMPermissionDataZonePortalReadWrite` which will enable associated accounts to execute Amazon DataZone APIs and access the data portal or you can choose `AWSRAMPermissionDataZoneDefault`, whcih will allow associated accounts to only execute Amazon DataZone APIs and will not provide data portal access. Amazon DataZone then creates a resource share in the AWS Resource Access Manager on your account's behalf, with the entered account ID(s) as principals.
6. You must notify the owner of the other AWS account(s) to accept your request. Invitations expire after seven (7) days.

Provide account access to your customer-managed KMS key

Amazon DataZone domains and their metadata are encrypted, either (by default) using a key held by AWS, or (optionally) a customer-managed key from AWS Key Management Service (KMS) that you own and provide during domain creation. If your domain is encrypted with a customer-managed key, then follow the procedure below to give the associated account permission to use the KMS key.

1. Sign in to the AWS Management Console and open the KMS console at https://console.aws.amazon.com/kms/.
2. To view the keys in your account that you create and manage, in the navigation pane choose **Customer managed keys**.

3. To view the keys in your account that you create and manage, in the navigation pane choose Customer managed keys.
4. In the list of KMS keys, choose the alias or key ID of the KMS key that you want to examine.
5. To allow or disallow external AWS accounts to use the KMS key, use the controls in the **Other AWS accounts** section of the page. IAM principals in these accounts (with proper KMS permissions themselves) can use the KMS key in cryptographic operations, such as encrypting, decrypting, re-encrypting, and generating data keys.

Accept an account association request from an Amazon DataZone domain and enable an environment blueprint

To accept association in the Amazon DataZone management console with an Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions.

Complete the following to accept association with an Amazon DataZone domain.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose **View requests** and select the inviting domain from the list. The state of the invitation should be **Requested**. Choose **Review request**.
3. Choose whether to enable the default data lake and/or data warehouse environment blueprints by selecting neither, both, or one of the boxes. You can do this later.
   - The data lake environment blueprint enables domain users to create and manage AWS Glue, Amazon S3, and Amazon Athena resources to publish and consume from a data lake.
   - The data warehouse environment blueprint enables domain users to create and manage Amazon Redshift resources to publish and consume from a data warehouse.
4. If you choose to select one or both of the default environment blueprints, then configure the following permissions and resources.
   - The **Manage access IAM role** provides permissions to Amazon DataZone to enable domain users to ingest and manage access to tables, like AWS Glue and Amazon Redshift. You can choose to have Amazon DataZone create and use a new IAM role, or you can choose from a list of existing IAM roles.

- The **Provisioning IAM role** provides permissions to Amazon DataZone to enable domain users to create and configure environment resources, like AWS Glue databases. You can choose to have Amazon DataZone create and use a new IAM role, or you can choose from a list of existing IAM roles.
- The **Amazon S3 bucket for Data Lake** is the bucket or path that Amazon DataZone will use when domain users store data lake data. You can use the default bucket selected by Amazon DataZone or choose your own existing Amazon S3 path by entering its path string. If you select your own Amazon S3 path, you will need to update IAM policies to provide Amazon DataZone with permissions to use it.

5. When you are satisfied with your configurations, choose **Accept and configure association**.

Reject an account association request from an Amazon DataZone domain

To reject an association request in the Amazon DataZone management console from an Amazon DataZone domain, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions.

Complete the following to reject an association request from an Amazon DataZone domain.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose **View requests** and select the inviting domain from the list. The state of the invitation should be **Requested**. Choose **Reject association**. Confirm your choice by choosing **Reject association**.

Enable an environment blueprint in an associated AWS account

To enable an environment blueprint in the Amazon DataZone management console, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions.

Complete the following to enable a blueprint in an associated domain.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Open the left navigation panel and choose **Associated domains**.
3. Choose the domain for which you want to enable an environment blueprint.
4. From the **Blueprints** list, choose either the **DefaultDataLake** or the **DefaultDataWarehouse**, or the **Amazon SageMaker**, or the **Custom AWS Service** blueprint.

> **Note**
>
> If you are enabling the **Custom AWS service** blueprint, you do not need to specify a manage access role. The permissions and the authorization mechanism for the **Custom AWS service** bluerpint are handled when you're creating environments using this blueprint. For more information, see Create an environment using a custom AWS service blueprint.

5. On the chosen blueprint's details page, choose **Enable in this account**.
6. On the Permissions and resources page, specify the following:

- If you're enabling the **DefaultDataLake** blueprint, for **Glue Manage Access role**, specify a new or existing service role that grants Amazon DataZone authorization to ingest and manage access to tables in AWS Glue and AWS Lake Formation.
- If you're enabling the **DefaultDataWarehouse** blueprint, for **Redshift Manage Access role**, specify a new or existing service role that grants Amazon DataZone authorization to ingest and manage access to datashares, tables and views in Amazon Redshift.
- If you're enabling the **Amazon SageMaker** blueprint, for **SageMaker Manage Access role**, specify a new or existing service role that grants Amazon DataZone permissions to publish Amazon SageMaker data to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to Amazon SageMaker published assets in the catalog.

> **Important**
>
> When you're enabling the **Amazon SageMaker** blueprint, Amazon DataZone checks whether the following IAM roles for Amazon DataZone exist in the current account and region. If these roles do not exist, Amazon DataZone automatically creates them.
>
> - AmazonDataZoneGlueAccess-<region>-<domainId>

- AmazonDataZoneRedshiftAccess-<region>-<domainId>

- For **Provisioning role**, specify a new or existing service role that grants Amazon DataZone authorization to create and configure environment resources using AWS CloudFormation in the environment account and region.
- If you're enabling the **Amazon SageMaker** blueprint, for the **Amazon S3 bucket for SageMaker-Glue data source**, specify an Amazon S3 bucket that is to be used by all SageMaker environments in the AWS account. The bucket prefix that you specify must be one of the following:
  - amazon-datazone*
  - datazone-sagemaker*
  - sagemaker-datazone*
  - DataZone-Sagemaker*
  - Sagemaker-DataZone*
  - DataZone-SageMaker*
  - SageMaker-DataZone*

7. Choose **Enable blueprint**.

Once you enable the chose blueprint(s), you can control which projects can use the blueprint(s) in your account to create environment profiles. You can do this by assigning managing projects to the blueprint’s configuration.

Specify managing projects on enabled DefaultDataLake or DefaultDataWarehouse blueprint

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Open the left navigation panel and choose **Associated domains** and then choose the domain where you want to add managing projects.
3. Choose the **Blueprints** tab and then choose DefaultDataLake or DefaultDataWareshouse blueprint.
4. By default, all projects within the domain can use the DefaultDataLake or DefaultDataWareshouse blueprint in the account to create environment profiles. However, you can restrict this by assigning managing projects to the blueprint. To add managing projects,

choose **Select managing project**, then choose the projects that you want to add as managing projects from the drop down menu, and then choose **Select managing projects(s)**.

Once you enable the DefaultDataWarehouse blueprint in your AWS account, you can add parameter sets to the blueprint configuration. A parameter set is a group of keys and values, required for Amazon DataZone to establish a connection to your Amazon Redshift cluster and is used to create data warehouse environments. These parameters include the name of your Amazon Redshift cluster, database, and the AWS secret that holds credentials to the cluster.

Adding parameter sets to the DefaultDataWarehouse blueprint

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Open the left navigation panel and choose **Associated domains** and then choose the domain where you want to add parameter sets.
3. Choose the **Blueprints** tab and then choose the DefaultDataWareshouse blueprint to open the blueprint details page.
4. Under the **Parameter sets** tab on the blueprint details page, choose **Create parameter set**.

   - Provide a Name for the parameter set.
   - Optionally, provide a description for the parameter set.
   - Select a region
   - Select either Amazon Redshift cluster or Amazon Redshift Serverless.
   - Select the AWS secret ARN that holds the credentials to the selected Amazon Redshift cluster or the Amazon Redshift Serverless workgroup. The AWS secret must be tagged with the `AmazonDataZoneDomain : [Domain_ID]` tag in order to be eligible for use within a parameter set.
     - If you do not have an existing AWS secret, you can also create a new secret by choosing **Create New AWS Secret**. This opens a dialog box where you can provide the name of the secret, username, and password. Once you choose **Create New AWS Secret**, Amazon DataZone creates a new secret in the AWS Secrets Manager service and ensures that the secret is tagged with the domain in which you are trying to create the parameter set.
   - Select either Amazon Redshift cluster or Amazon Redshift Serverless workgroup.
   - Enter the name of the database within the selected Amazon Redshift cluster or Amazon Redshift Serverless workgroup.

- Choose **Create parameter set**.

Once you enable the Amazon SageMaker blueprint in your AWS account, you can add parameter sets to the blueprint configuration. A parameter set is a group of keys and values, required for Amazon DataZone to establish a connection to your Amazon SageMaker and is used to create sagemaker environments.

Adding parameter sets to the Amazon SageMaker blueprint

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Choose **View domains** and then choose the domain that contains the enabled blueprint where you want to add the parameter set.
3. Choose the **Blueprints** tab and then choose the Amazon SageMaker blueprint to open the blueprint's details page.
4. Under the **Parameter sets** tab on the blueprint details page, choose **Create parameter set**, and then specify the following:

    - Provide a **Name** for the parameter set.
    - Optionally, provide a **Description** for the parameter set.
    - Specify the Amazon SageMaker domain authentication type. You can choose either IAM or IAM Identity Center (SSO).
    - Specify an AWS region.
    - Specify an AWS KMS key for data encryption. You can choose an existing key or create a new key.
    - Under **Environment parameters**, specify the following:
        - VPC ID - the ID that you're using for the VPC of the Amazon SageMaker environment. You can specify an existing or create a new VPC.
        - Subnets - one or more IDs for a range of IP addresses for specific resources within your VPC.
        - Network access - choose either **VPC only** or **Public internet only**.
        - Security group - the security group to use when configuring VPC and subnets.
    - Under Data source parameters, choose one of the following:
        - AWS Glue only

- AWS Glue + Amazon Redshift Serverless. If you choose this option, specify the following:
  - Specify the AWS secret ARN that holds the credentials to the selected Amazon Redshift cluster. The AWS secret must be tagged with the `AmazonDataZoneDomain : [Domain_ID]` tag in order to be eligible for use within a parameter set.

    If you do not have an existing AWS secret, you can also create a new secret by choosing **Create New AWS Secret**. This opens a dialog box where you can provide the name of the secret, username, and password. Once you choose **Create New AWS Secret**, Amazon DataZone creates a new secret in the AWS Secrets Manager service and ensures that the secret is tagged with the domain in which you are trying to create the parameter set.
  - Specify the Amazon Redshift workgroup you want to use when creating environments.
  - Specify the name of the database (within the workgroup you've chosen) that you want to use when creating environments.
- AWS Glue only + Amazon Redshift Cluster
  - Specify the AWS secret ARN that holds the credentials to the selected Amazon Redshift cluster. The AWS secret must be tagged with the `AmazonDataZoneDomain : [Domain_ID]` tag in order to be eligible for use within a parameter set.

    If you do not have an existing AWS secret, you can also create a new secret by choosing **Create New AWS Secret**. This opens a dialog box where you can provide the name of the secret, username, and password. Once you choose **Create New AWS Secret**, Amazon DataZone creates a new secret in the AWS Secrets Manager service and ensures that the secret is tagged with the domain in which you are trying to create the parameter set.
  - Specify the Amazon Redshift cluster you want to use when creating environments.
  - Specify the name of the database (within the cluster you've chosen) that you want to use when creating environments.

5. Choose **Create parameter set**.

Add Amazon SageMaker as a trusted service in the associated AWS account

If you've enabled the Amazon SageMaker blueprint, you must also add SageMaker as one of the trusted services within Amazon DataZone. To do this, complete the following procedure:

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Choose **View domains** and then choose the domain that contains the enabled SageMaker blueprint.
3. Choose the **Trusted services**, then choose the **Amazon SageMaker**, and then choose **Enable**.

Remove an associated account

To remove an associated AWS account in the Amazon DataZone management console, you must assume an IAM role in the account with administrative permissions. Configure the IAM permissions required to use the Amazon DataZone management console to obtain the minimum permissions.

Complete the following procedure to remove an associated account from your domain.

1. Sign in to the AWS Management Console and open the Amazon DataZone management console at https://console.aws.amazon.com/datazone.
2. Choose **View Domains** and choose the domain's name from the list. The name is a hyperlink.
3. Scroll down to the **Associated accounts** tab. Choose the account ID for the AWS account you want to remove.
4. Choose **Disassociate**. Confirm your choice by entering disassociate in the field and choosing **Disassociate**.
5. The account is now removed from your domain and cannot be used by the domain's users to publish and consume data.

Working with the Amazon DataZone data catalog

You can use the Amazon DataZone business data catalog to catalog data across your organization with business context and thus enable everyone in your organization to find and understand data quickly. For more information, see Amazon DataZone terminology and concepts.

Topics

Create, edit, or delete a business glossary

In Amazon DataZone, a business glossary is a collection of business terms (words) that may be associated with assets (data). It provides appropriate vocabularies with a list of business terms and their definitions for business users to ensure the same definitions are used across the organization when analyzing data. Business glossaries are created in the catalog domain and can be applied to assets and columns to help understand key characteristics of that asset or column. One or more glossary terms can be applied. A business glossary can be a flat list of terms where any term in the business glossary can be associated with a sublist of other terms. For more information, see Amazon DataZone terminology and concepts. To create, edit, or delete a glossary in your Amazon DataZone domain, you must be the member of the owning project with the right permissions for that domain.

To create a glossary, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Glossaries**, and then choose **Create glossary**.
4. Specify a name, description, owner for the glossary and then choose **Create glossary**.
5. Enable the new glossary by choosing the **Enabled** toggle.

6. On the glossary's details page, you can choose **Create readme** to add some additional information about this glossary.

To disable or enable a business glossary, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Glossaries**, and locate the business glossary that you want to disable/enable.
4. On the glossary details page, locate the **Enable/Disable** toggle and use it to enable or disable your selected glossary.

> **Note**
>
> Disabling a glossary also disables all the terms that it contains.

To edit a business glossary, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Glossaries**, and locate the business glossary that you want to edit.
4. On the glossary details page, expand **Actions** and then choose **Edit** to edit the glossary.
5. Make your updates to the name, description, and then choose **Save**.

To delete a business glossary, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data

portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.

2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Glossaries**, and locate the business glossary that you want to delete.
4. On the glossary details page, expand **Actions** and then choose **Delete** to delete the glossary.

> **Note**
>
> You must delete all existing terms in the glossary before you can delete the glossary.

5. Confirm the deletion of the glossary by choosing **Delete**.

Create, edit, or delete a term in a glossary

In Amazon DataZone, a business glossary is a collection of business terms that may be associated with assets (data). For more information, see Amazon DataZone terminology and concepts. To create, edit, or delete terms in a glossary in your Amazon DataZone domain, you must be the member of the owning project with the right permissions for that domain.

In Amazon DataZone, business glossary terms can have close descriptions. To set the context of a particular term, you can specify relationships among terms. When you define a relationship for a term, it is automatically added to the definition of the related term. The glossary term relationships available in Amazon DataZone include the following:

- **Is a Type of** - indicates that the current term is a type of the identified term. Indicates that the identified term is a parent to the current term.
- **Has Types** - indicates that the current term is a generic term for the indicated specific term or terms. This relationship can denote child terms for the generic term.

To create a new term, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you're an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.

2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Glossaries**, and then choose the glossary where you want to create the new term.
4. Specify a name, description, owner for the term and then choose **Create term**.
5. Enable the new term by choosing the **Enabled** toggle.
6. To add **Readme**, navigate to the term details page, and then you can choose **Create readme** to add some additional information about this glossary.
7. To add relationships, navigate to the term details page, choose **Term Relationships** section, and then choose **Add Glossary Terms**. In the dialog, choose the relationship and the terms you want to relate, and then choose **Close** to add a term to the appropriate relationship type. This relationship is also added to all the terms you made related.

To edit a term in a glossary, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you're an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Glossaries**, locate the glossary that contains the term that you you want to edit, and then choose that term.
4. On the term details page, expand **Actions** and then choose **Edit** to edit the term.
5. Make your updates to the name, description , and then choose **Save**.

To delete a term in a glossary, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you're an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Glossaries**, locate the glossary that contains the term that you you want to delete, and then choose that term.
4. On the glossary details page, expand **Actions** and then choose **Delete** to delete the term.

5. Confirm the deletion of the term by choosing **Delete**.

Create, edit, or delete metadata forms

In Amazon DataZone, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. It serves as an extensible mechanism for data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve a mechanism to enforce consistency to all assets being published to the Amazon DataZone catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information, see Amazon DataZone terminology and concepts. To create, edit, or delete metadata forms in your Amazon DataZone domain, you must be a member of the owning project who has the right credentials.

To create a metadata form, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Metadata forms** and then choose **Create form**.
4. Specify the metadata form name, description, owner and then choose **Create form**.

To edit a metadata form, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Metadata forms**, and then locate the metadata form that you want to edit.
4. On the metadata form's details page, expand **Actions**, and then choose **Edit**.

5. Perform your updates to the name, description, owner fields, and then choose **Update form**.

To delete a metadata form, complete the following steps:

> **Note**
>
> Before you can delete a metadata form, you must remove it from all asset types or assets to which it is applied.

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Metadata forms**, and then locate the metadata form that you want to delete.
4. If the metadata form that you want to delete is enabled, disable the metadata form by choosing the **Enabled** toggle.
5. On the metadata form's details page, expand **Actions**, and then choose **Delete**.
6. Confirm deletion by choosing **Delete**.

Create, edit, or delete fields in metadata forms

In Amazon DataZone, metadata forms are simple forms to augment additional business context to the asset metadata in the catalog. It serves as an extensible mechanism for data owners to enrich the asset with information that can help data users when they search and find that data. Metadata forms can also serve an mechanism to enforce consistency to all assets being published to the Amazon DataZone catalog.

A metadata form definition is composed of one or more field definitions, with support for boolean, date, decimal, integer, string, and business glossary field value data types. For more information, see Amazon DataZone terminology and concepts. To create, edit, or delete fields in metadata forms in your Amazon DataZone domain, you must be a member of the owning project who has the right credentials.

To create a field in a metadata form, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Metadata forms** and then choose the metadata form where you want to create field(s).
4. On the form's details page, choose **Create field**.
5. Specify the field name, description, type, and whether this is a required field, and then choose **Create field**.

To edit a field in a metadata form, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.
3. In the Amazon DataZone Data Portal, choose **Metadata forms** and then choose the metadata form where you want to edit field(s).
4. On the form's details page, choose the field that you want to edit, then expand **Actions**, and choose **Edit**.
5. Make your updates to the field name, description, type, and whether this is a required field, and then choose **Update field**.

To delete a field in a metadata form, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you’re an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Navigate to the **Catalog** menu in the top navigation bar next to **Search**.

3. In the Amazon DataZone Data Portal, choose **Metadata forms** and then choose the metadata form where you want to delete field(s).
4. On the form's details page, choose the field that you want to delete, then expand **Actions**, and choose **Delete**.
5. Confirm deletion by choosing **Delete**.

Working with projects and environments in Amazon DataZone

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and consuming data assets in the Amazon DataZone catalog. Each Amazon DataZone project has a set of access controls applied to it so that only authorized individuals, groups, and roles can access the project and the data assets that this project subscribes to, and can use only those tools that are defined by the project permissions. Projects act as an identity principal that receives access grants to underlying resources, enabling Amazon DataZone to operate within an organization's infrastructure without relying on individual user's credentials. For more information, see Amazon DataZone terminology and concepts

Topics

Create an environment profile

In Amazon DataZone, an environment profile is a template that you can use to create environments. The purpose of an environment profile is to simplify environment creation by embedding placement information such as AWS account and region within the profiles. For more information, see Amazon DataZone terminology and concepts. To create environment profiles in an Amazon DataZone domain, you must belong to an Amazon DataZone project. All environment profiles are owned by projects and can be used by all authorized users, from any project, to create new environments.

To create an environment profile

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you're an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Within the data portal, choose **Browse projects** and select the project in which you want to create the environment profile.
3. Navigate to the **Environments** tab within the project, then choose **Create environment profile**.
4. Configure the following fields:

   - **Name –** The name for your environment profile.
   - **Description –** (Optional) A description for your environment profile.
   - **Owner Project** - The project where the profile is being created is selected by default in this field.
   - **Blueprint –** The blueprint for which this profile is created. You can choose one of the default Amazon DataZone blueprints (Data Lake or Data Warehouse).

     If you specified the Data Warehouse blueprint, do the following:

     - Provide a parameter set. To select an existing parameter set choose the option **Choose a parameter set**. If you want to enter your own parameters, choose **Enter my own**.
     - If you choose to select an existing parameter, then do the following:
       - Select an AWS account from the drop down.
       - Select a parameter set from the dropdown.
     - If you choose to enter your own parameters, do the following:
       - Provide the AWS parameters by selecting the AWS Account and Region from the dropdown.
       - Provide Redshift Data Wareshoue parameters:
         - Select either Amazon Redshift cluster or Amazon Redshift Serverless

  - Enter the AWS Secret ARN that holds the credentials to the selected Amazon Redshift cluster or Amazon Redshift Serverless workgroup. The AWS secret must be tagged with the domain Id and Project Id where you are creating the environment profile.
    - `AmazonDataZoneDomain: [Domain_ID]`
    - `AmazonDataZoneProject: [Project_ID]`
  - Enter the name of Amazon Redshift cluster or Amazon Redshift Serverless workgroup.
  - Enter the name of the database within the selected Amazon Redshift cluster or Amazon Redshift Serverless workgroup.
- In the **Authorized projects** section, specify the projects that can use the environment profile for creating environments. By default, all projects within the domain can use the environment profiles in the account to create environments. To keep this default setting, choose **All projects**. However, you can restrict this by assigning authorized projects to the environment. To do so, choose **Authorized projects only** and then specify projects that can use this project profile to create environments.
- In the **Publishing** section, either choose one of the following options:
  - **Publish from any schema**: If you choose this option, environments created using this environment profile can be used to publish from any schema within database selected in the Redshift parameters provided above. Users of the environment created using this environment profiles can also provide their own Amazon Redshift parameters to publish from any schema within the AWS account and region selected in the environment profile.
  - **Publish from only default environment schema**: If you choose this option, environments created using this can be used to publish only from the default schema created by Amazon DataZone for that environment. Users of the environment created using this environment profiles cannot provide their own Amazon Redshift parameters.
  - **Don’t allow publishing**: If you choose this option, environments created using this environment profile can only be used for subscribing and consumption of data. Environments cannot be used to publish any data at all.

If you specified the Data Lake blueprint, do the following:

- In the **AWS account parameters** section, specify the AWS account number and the AWS account region where the potential environments will be created.

  - In the **Authorized projects** section, specify the projects that can use the environment profile with the built-in Data Lake environment profile for creating environments. By default, all projects within the domain can use the data lake blueprint in the account to create environment profiles. To keep this default setting, choose **All projects**. However, you can restrict this by assigning projects to the blueprint. To do so, choose **Authorized projects** only and then specify projects that can use this project profile to create environments.
  - In the **Databases** section, either choose **Any database** to enable publishing from any database within the AWS account and region where the environment is created or choose Only default database to enable publishing from only the default publishing database that is created with the environment.

5. Choose **Create environment profile**.

Edit an environment profile

In Amazon DataZone, an environment profile is a template that you can use to create environments. For more information, see Amazon DataZone terminology and concepts. To edit an existing environment profiles in an Amazon DataZone domain, you must belong to an Amazon DataZone project.

To edit an environment profile

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Within the data portal, choose **Browse projects** and select the project in which you want to edit the environment profile.
3. Navigate to the **Environments** tab within the project, then choose **Environment profiles**, and then choose the environment profile that you want to edit.

   If you are editing a Data Warehouse environment profile, you can only edit the name and the description of an existing environment profile.

If you are editing a Data Lake environment profile, you can edit the name and the description of the profile and you can also edit the projects that are authorized to use this profile to create environments and you can edit databases. To edit these settings, do the following:

- In the **Authorized projects** section, specify the projects that can use the environment profile with the built-in Data Lake environment profile for creating environments. By default, all projects within the domain can use the data lake blueprint in the account to create environment profiles. To keep this default setting, choose **All projects**. However, you can restrict this by assigning projects to the blueprint. To do so, choose **Authorized projects** only and then specify projects that can use this project profile to create environments.
- In the **Databases** section, either choose **Any database** to enable publishing from any database within the AWS account and region where the environment is created or choose Only default database to enable publishing from only the default publishing database that is created with the environment.

When you complete your edits, choose **Edit environment profile**.

Delete an environment profile

In Amazon DataZone, an environment profile is a template that you can use to create environments. The purpose of an environment profile is to simplify environment creation by embedding placement information such as AWS account and region within the profiles. For more information, see Amazon DataZone terminology and concepts. To delete environment profiles in an Amazon DataZone domain, you must belong to an Amazon DataZone project.

> **Note**
>
> When you delete an environment profile, you can't create any more environments using this profile.

To delete an environment profile

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the

Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.

2. Within the data portal, choose **Browse projects** and select the project in which you want to delete the environment profile.
3. Navigate to the **Environments** tab within the project, then choose **Environment profiles**, and then choose the environment profile that you want to delete.
4. Select the environment profile you want to delete, then choose **Actions**, **Delete** and confirm deletion.

Create a new environment

In Amazon DataZone projects, environments are collections of configured resources (for example, an Amazon S3 bucket, an AWS Glue database, or an Amazon Athena workgroup), with a given set of IAM principals (environment user roles) with assigned owner or contributor permissions who can operate on those resources. For more information, see Amazon DataZone terminology and concepts.

Any Amazon DataZone user with the required permissions to access the data portal can create an Amazon DataZone environment within a project.

To create a new environment, complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Browse all projects** and select the project in which you want to create a new environment.
3. Choose **Create environment**, specify values for the following fields, and then choose **Create environment**:

   - **Name** – the environment name
   - **Description** – a description of the environment
   - **Environment profile** – choose an existing environment profile or create a new one. An environment profile is a template that you can use to create environments. For more information, see Amazon DataZone terminology and concepts.

Once you've selected the environment profile, under the **Parameters** section, specify the values for the fields that are part of this environment profile.

Edit an environment

In Amazon DataZone projects, environments are collections of configured resources (for example, an Amazon S3 bucket, an AWS Glue database, or an Amazon Athena workgroup), with a given set of IAM principals (with assigned contributor permissions) who can operate on those resources. For more information, see Amazon DataZone terminology and concepts.

Any Amazon DataZone user with the required permissions to access the data portal can edit an Amazon DataZone environment within a project.

To edit an existing environment, complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Browse projects** from the top navigation pane and select the project that contains the environment that you want to edit.
3. Locate and choose the environment to open its details page. Then expand **Actions** and choose **Edit environment**.
4. Make your edits to the environment's name and description, and then choose **Save changes**.

Delete an environment

In Amazon DataZone projects, environments are collections of configured resources (for example, an Amazon S3 bucket, an AWS Glue database, or an Amazon Athena workgroup), with a given set of IAM principals (with assigned contributor permissions) who can operate on those resources. For more information, see Amazon DataZone terminology and concepts.

Any Amazon DataZone user with the required permissions to access the data portal can delete an Amazon DataZone environment within a project.

To delete an existing environment, complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Browse project** from the top navigation pane and select the project that contains the environment that you want to delete.
3. Locate and choose the environment to open its details page, then expand **Actions** and choose **Delete environment**.
4. In the **Delete environment** pop up window, confirm deletion by typing `Delete` in the field and then choose **Delete environment**.

   You can successfully delete an environment only after all entities with a dependency to this environment have been deleted. To delete an environment, you must first delete all its associated data sources and subscription targets.

Create a new project

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and consuming data assets in the Amazon DataZone catalog. For more information, see Amazon DataZone terminology and concepts.

Any Amazon DataZone user with the required permissions to access the data portal can create an Amazon DataZone project.

To create a new project complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. In the Amazon DataZone data portal, choose **Create Project**.
3. Specify values for the following fields, and then choose **Create project**:

   - **Name –** The project name.
   - **Description –** A description of the project.

Edit project

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and consuming data assets in the Amazon DataZone catalog. For more information, see Amazon DataZone terminology and concepts. To edit an Amazon DataZone project, you must be the owner of that project or the domain administrator of the domain that contains this project.

To edit an existing project, complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Browse projects**.
3. Choose the project that you want to edit. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Find project** field.
4. Expand **Actions** and choose **Edit project**.
5. Perform your updates to the project name and description and then choose **Save**.

Delete project

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and/or consuming data assets in the Amazon DataZone catalog. For more information, see Amazon DataZone terminology and concepts.

The act of deleting a project is final. Deletion irrevocably deletes the project's contents, including data sources, environments, assets, glossaries, and metadata forms. Amazon DataZone revokes grants Amazon DataZone has placed on managed assets via Lake Formation and Amazon Redshift. Deleting a project does not delete non-Amazon DataZone AWS resources that Amazon DataZone may have helped you create. If you no longer need these AWS resources, delete them in their respective AWS service and account.

To delete an Amazon DataZone project, you must be an owner of the project.

To delete an existing project, complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. An IAM principal can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Browse projects** from the top navigation pane.
3. Choose the project that you want to delete. If you don't see it in the list of projects, you can search for it by specifying the project name in the **Find project** field.
4. Expand **Actions** and choose **Delete project**.

   Review the informational warnings about the potential impact of deleting the project.
5. If you accept the warnings, then type in the confirmation text, and choose **Delete**.

> ⚠ **Important**
>
> Deleting a project is an irrevocable action that cannot be undone by you or by AWS.

> ⓘ **Note**
>
> When you or your domain users create an environment in a project, Amazon DataZone creates AWS resources in your domain or associated accounts to provide you and your domain users with functionality. Below is the list of AWS resources that Amazon DataZone may create for a project, along with the default name. Deleting a project does not delete any of these AWS resources in your AWS accounts.
>
> - IAM roles: datazone_usr_<environmentId>.
> - Glue databases: (1) <environmentName>_pub_db-*, (2) <environmentName>_sub_db-*. If there was already an existing database of this name, Amazon DataZone will add the environment ID.
> - Athena workgroups: <environmentName>-*. If there was already an existing workgroup of this name, Amazon DataZone will add the environment ID.
> - CloudWatch log group: datazone_<environmentId>

Leave project

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and consuming data assets in the Amazon DataZone catalog. For more information, see Amazon DataZone terminology and concepts.

To leave an existing project, complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project.
3. Choose the project that you want to leave. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Find project** field.
4. Expand **Actions** and choose **Leave project**.

Add members to a project

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and consuming data assets in the Amazon DataZone catalog. For more information, see Amazon DataZone terminology and concepts.

You must be a project owner or contributor to add members to a project. You can add SSO groups, SSO users, or IAM principals (roles or users) as project members.

To add members to an exiting project, complete the following steps.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project.
3. Choose the project to which you want to add memebrs. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Find project** field.
4. On the project's details page, select the **Members** tab and the choose **All members** node.

5. In the project Members tab, choose **Add members**.
6. In the **Add members to project** pop up window, specify the user(s) that you want to add and specify their role within the project (owner or contributor) and then choose **Add members**.

> **Note**
>
> You can add an IAM principal as a project member if that principal already has a Amazon DataZone user profile in the domain. Amazon DataZone automatically creates a user profile for an IAM principal when it successfully interacts with the domain via the portal, API, or CLI. You cannot create a user profile for an IAM principal. To add IAM principals as project members in the case where the IAM principal does not have an existing Amazon DataZone user profile in the domain, ask your administrator to add the following two IAM permissions to your domain's **AmazonDataZoneDomainExecutionRole** in the IAM console: `iam:GetUser` and `iam:GetRole`. Separately, to perform actions in the domain, the IAM principal must have the corresponding IAM permissions to such actions.

Remove members from a project

In Amazon DataZone, projects enable a group of users to collaborate on various business use cases that involve publishing, discovering, subscribing to, and consuming data assets in the Amazon DataZone catalog. For more information, see Amazon DataZone terminology and concepts. You must be a project owner in order to remove members from a project.

To remove members from an exiting project, complete the following steps.

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you're an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. Choose **Select project** from the top navigation pane and select the project.
3. Choose the project where you want to remove memebrs. If you don't readily see it in the list of projects, you can search for it by specifying the project name in the **Find project** field.
4. On the project's details page, select the **Members** tab and the choose **All members** node.
5. In the project Members tab, choose the member(s) that you want to remove from the project and then choose **Remove**.

6. In the **Remove members** pop up window, confirm removal by choosing **Remove members**.

Creating inventory and publishing data in Amazon DataZone

This section describes the tasks and procedures that you want to perform in order to create an inventory of your data in Amazon DataZone and to publish your data in Amazon DataZone.

In order to use Amazon DataZone to catalog your data, you must first bring your data (assets) as inventory of your project in Amazon DataZone. Creating inventory for a particular project, makes the assets discoverable only to that project’s members. Project inventory assets are not available to all domain users in search/browse unless explicitly published. After creating a project inventory, data owners can curate their inventory assets with the required business metadata by adding or updating business names (asset and schema), descriptions (asset and schema), read me, glossary terms (asset and schema), and metadata forms.

The next step of using Amazon DataZone to catalog your data, is to make your project’s inventory assets discoverable by the domain users. You can do this by publishing the inventory assets to the Amazon DataZone catalog. Only the latest version of the inventory asset can be published to the catalog and only the latest published version is active in the discovery catalog. If an inventory asset is updated after it's been published into the Amazon DataZone catalog, you must explicitly publish it again in order for the latest version to be in the discovery catalog.

Topics

- Asset revisions in Amazon DataZone
- Data quality in Amazon DataZone
- Using machine learning and generative AI
- Data lineage in Amazon DataZone (Preview)

Configure Lake Formation permissions for Amazon DataZone

When you create an environment using the built-in data lake blueprint (**DefaultDataLake**), an AWS Glue database is added in Amazon DataZone as part of this environment's creation process. If you want to publish assets from this AWS Glue database, no additional permissions are needed.

However, if you want to publish assets and subscribe to assets from an AWS Glue database that exists outside of your Amazon DataZone environment, you must explicitly provide Amazon DataZone with the permissions to access tables in this external AWS Glue database. To do this, you must complete the following settings in AWS Lake Formation and attach necessary Lake Formation permissions to the AmazonDataZoneGlueAccess-<region>-<domainId> .

- Configure the Amazon S3 location for your data lake in AWS Lake Formation with **Lake Formation** permission mode or **Hybrid access mode**. For more information, see https://docs.aws.amazon.com/lake-formation/latest/dg/register-data-lake.html.
- Remove the `IAMAllowedPrincipals` permission from the Amazon Lake Formation tables for which Amazon DataZone handles permissions. For more information, see https://docs.aws.amazon.com/lake-formation/latest/dg/upgrade-glue-lake-formation-background.html.
- Attach the following AWS Lake Formation permissions to the AmazonDataZoneGlueAccess-<region>-<domainId>:
  - `Describe` and `Describe grantable` permissions on the database where the tables exist
  - `Describe`, `Select`, `Describe Grantable`, `Select Grantable` permissions on the all the tables in the above database that you want DataZone to manage access on your behalf.

> **Note**
>
> Amazon DataZone supports the AWS Lake Formation Hybrid mode. Lake Formation hybrid mode enables you to start managing permissions on you AWS Glue databases and tables through Lake Formation, while continuing to maintain any existing IAM permissions on

these tables and databases. For more information, see Amazon DataZone integration with AWS Lake Formation hybrid mode

For more information, see Troubleshooting AWS Lake Formation permissions for Amazon DataZone.

Amazon DataZone integration with AWS Lake Formation hybrid mode

Amazon DataZone is integrated with AWS Lake Formation hybrid mode. This integration enables you to easily publish and share your AWS Glue tables through Amazon DataZone without the need to register them in AWS Lake Formation first. Hybrid mode allows you to start managing permissions on your AWS Glue tables through AWS Lake Formation while continuing to maintain any existing IAM permissions on these tables.

To get started, you can enable the **Data location registration** setting under the **DefaultDataLake** blueprint in the Amazon DataZone management console.

Enable integration with AWS Lake Formation hybrid mode

1. Navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with your account credentials.
2. Choose **View domains** and choose the domain where you want to enable the integration with AWS Lake Formation hybrid mode.
3. On the domain details page, navigate to the **Blueprints** tab.
4. From the **Blueprints** list, choose the **DefaultDataLake** blueprint.
5. Make sure that the DefaultDataLake blueprint is enabled. If it's not enabled, follow the steps in Enable built-in blueprints in the AWS account that owns the Amazon DataZone domain to enable it in your AWS Account.
6. On the DefaultDataLake details page, open the **Provisioning** tab and choose the **Edit** button in the top right corner of the page.
7. Under **Data location registration**, check the box to enable the data location registration.
8. For the data location management role, you can create a new IAM role or select an existing IAM role. Amazon DataZone uses this role to manage read/write access to the chosen Amazon S3 bucket(s) for Data Lake using AWS Lake Formation hybrid access mode. For more information, see AmazonDataZoneS3Manage-<region>-<domainId>.

9. Optionally, you can choose to exclude certain Amazon S3 locations if you do not want Amazon DataZone to automatically register them in hybrid mode. For this, complete the following steps:

    - Choose the toggle button to exclude specified Amazon S3 locations.
    - Provide the URI of the Amazon S3 bucket you want to exclude.
    - To add additional buckets, choose **Add S3 location.**

    > **Note**
    >
    > Amazon DataZone only allows excluding a root S3 location. Any S3 locations within the path of a root S3 location will be automatically excluded from registration.

    - Choose **Save changes**.

Once you have enabled the data location registration setting in your AWS account, when a data consumer subscribes to an AWS Glue table managed through IAM permissions, Amazon DataZone will first register the Amazon S3 locations of this table in hybrid mode, and then grant access to the data consumer by managing permissions on the table through AWS Lake Formation. This ensures that IAM permissions on the table continue to exist with newly granted AWS Lake Formation permissions, without disrupting any existing workflows.

How to handle encrypted Amazon S3 locations when enabling AWS Lake Formation hybrid mode integration in Amazon DataZone

If you are using an Amazon S3 location encrypted with an Customer managed or AWS Managed KMS key, the **AmazonDataZoneS3Manage** role must have the permission to encrypt and decrypt data with the KMS key, or the KMS key policy must grant permissions on the key to the role.

If your Amazon S3 location is encrypted with an AWS managed key, add the following inline policy to the **AmazonDataZoneDataLocationManagement** role:

```
 {
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt",
      "kms:ReEncrypt*",
      "kms:GenerateDataKey*",
      "kms:DescribeKey"
    ],
    "Resource": "<AWS managed key ARN>"
  }
]
```

If your Amazon S3 location is encrypted with a customer managed key, do the following:

1. Open the AWS KMS console at https://console.aws.amazon.com/kms and log in as an AWS Identity and Access Management (IAM) administrative user or as a user who can modify the key policy of the KMS key used to encrypt the location.
2. In the navigation pane, choose **Customer managed keys**, and then choose the name of the desired KMS key.
3. On the KMS key details page, choose the **Key policy** tab, and then do one of the following to add your custom role or the Lake Formation service-linked role as a KMS key user:

   - If the default view is showing (with Key administrators, Key deletion, Key users, and Other AWS accounts sections) – under the **Key users** section, add the **AmazonDataZoneDataLocationManagement** role.
   - If the key policy (JSON) is showing – edit the policy to add **AmazonDataZoneDataLocationManagement** role to the object "Allow use of the key," as shown in the following example

   ```
   ...
           {
               "Sid": "Allow use of the key",
               "Effect": "Allow",
               "Principal": {
                   "AWS": [
                       "arn:aws:iam::111122223333:role/service-role/
   AmazonDataZoneDataLocationManage-<region>-<domain-id>",
                       "arn:aws:iam::111122223333:user/keyuser"
                   ]
               },
   ```

```
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        },
        ...
```

Note

If the KMS key or Amazon S3 location are not in the same AWS account as the data catalog, follow the instructions in Registering an encrypted Amazon S3 location across AWS accounts.

Create custom asset types

In Amazon DataZone, assets represent specific types of data resources such as database tables, dashboards, or machine learning models. To provide consistency and standardization when describing catalog assets, an Amazon DataZone domain must have a set of asset types that define how assets are represented in the catalog. An asset type defines the schema for a specific type of asset. An asset type has a set of required and optional nameable metadata form types (for example, govForm or GovernanceFormType). Asset type in Amazon DataZone are versioned. When assets are created, they are validated against the schema defined by their asset type (typically latest version), and if an invalid structure is specified, asset creation fails.

System asset types - Amazon DataZone provisions service-owned system asset types (including GlueTableAssetType, GlueViewAssetType, RedshiftTableAssetType, RedshiftViewAssetType, and S3ObjectCollectionAssetType) and system form types (including DataSourceReferenceFormType, AssetCommonDetailsFormType, and SubscriptionTermsFormType). System asset types cannot be edited.

Custom asset types - for creating custom asset types, you start by creating the required metadata form types and glossaries to use in the form types. You can then create custom asset types by specifying name, description, and associated metadata forms which can be required or optional.

For asset types with structured data, to represent the column schema in the data portal, you can use the `RelationalTableFormType` to add the technical metadata to your columns, including column names, descriptions, and data types) and the `ColumnBusinessMetadataForm` to add the business descriptions of the columns, including business names, glossary terms, and custom key value pairs.

To create a custom asset type via the Data portal, complete the following steps:

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project where you want to create a custom asset type.
3. Navigate to the **Data** tab for the project.
4. Choose **Asset types** from the left navigation pane, then choose **Create asset type**.
5. Specify the following and then choose **Create**.
   - **Name** - the name of the custom asset type
   - **Description** - the description of the custom asset type.
   - **Choose Add metadata forms** to add metadata forms to this custom asset type.
6. Once the custom asset type is created, you can use it to create assets.

To create a custom asset type via the APIs, complete the following steps:

1. Create a metadata form type by invoking the `CreateFormType` API action.

   The following is an Amazon SageMaker example:

```
m_model = "

structure SageMakerModelFormType {
   @required
   @amazon.datazone#searchable
   modelName: String

   @required
   modelArn: String

   @required
   creationTime: String
}
"

CreateFormType(
    domainIdentifier="my-dz-domain",
    owningProjectIdentifier="d4bywm0cja1dbb",
    name="SageMakerModelFormType",
    model=m_model
    status="ENABLED"
    )
```

2. Next, you can create an asset type by invoking the `CreateAssetType` API action. You can create asset types only via Amazon DataZone APIs using the available system form types (`SubscriptionTermsFormType` in the below example) or your custom form types. For system form types, the type name must begin with `amazon.datazone`.

```
CreateAssetType(
    domainIdentifier="my-dz-domain",
    owningProjectIdentifier="d4bywm0cja1dbb",
    name="SageMakerModelAssetType",
    formsInput={
        "ModelMetadata": {
            "typeIdentifier": "SageMakerModelMetadataFormType",
            "typeRevision": 7,
            "required": True,
        },
        "SubscriptionTerms": {
            "typeIdentifier": "amazon.datazone.SubscriptionTermsFormType",
            "typeRevision": 1,
            "required": False,
        },
    },
)
```

The following is an example for creating an asset type for structured data:

```
CreateAssetType(
    domainIdentifier="my-dz-domain",
    owningProjectIdentifier="d4bywm0cja1dbb",
    name="OnPremMySQLAssetType",
    formsInput={
        "OnpremMySQLForm": {
            "typeIdentifier": "OnpremMySQLFormType",
            "typeRevision": 5,
            "required": True,
        },
        "RelationalTableForm": {
            "typeIdentifier": "RelationalTableFormType",
            "typeRevision": 1,
            "required": True,
        },
        "ColumnBusinessMetadataForm": {
            "typeIdentifier": "ColumnBusinessMetadataForm",
            "typeRevision": 1,
            "required": False,
        },
        "SubscriptionTerms": {
            "typeIdentifier": "SubscriptionTermsFormType",
            "typeRevision": 1,
            "required": False,
        },
    },
)
```

3. And now, you can create an asset using the custom asset types you created in the steps above.

```
CreateAsset(
   domainIdentifier="my-dz-domain",
   owningProjectIdentifier="d4bywm0cja1dbb",
   owningProjectIdentifier="my-project",
   name="MyModelAsset",
   glossaryTerms="xxx",
   formsInput=[{
        "formName": "SageMakerModelForm",
        "typeIdentifier": "SageMakerModelForm",
        "typeRevision": "5",
        "content": "{\n \"ModelName\" : \"sample-ModelName\",\n \"ModelArn\" :
 \"999999911111\"\n}"
        }
        ]
)
```

And in this example you're creating a structured data asset:

```
CreateAsset(
   domainIdentifier="my-dz-domain",
   owningProjectIdentifier="d4bywm0cja1dbb",
   name="MyModelAsset",
   glossaryTerms="xxx",
   formsInput=[{
        "formName": "RelationalTableForm",
        "typeIdentifier": "amazon.datazone.RelationalTableForm",
        "typeRevision": "1",
        "content": ".."
        },
        {
        "formName": "mySQLTableForm",
        "typeIdentifier": "mySQLTableForm",
        "typeRevision": "6",
        "content": ".."
        },
        {
        "formName": "mySQLTableForm",
        "typeIdentifier": "mySQLTableForm",
        "typeRevision": "1",
        "content": ".."
        },
        .....
        ]
)
```

Create and run an Amazon DataZone data source for the AWS Glue Data Catalog

In Amazon DataZone, you can create an AWS Glue Data Catalog data source in order to import technical metadata of database tables from AWS Glue. To add a data source for the AWS Glue Data Catalog, the source database must already exist in AWS Glue.

When you create and run an AWS Glue data source, you add assets from the source AWS Glue database to your Amazon DataZone project's inventory. You can run your AWS Glue data sources on a set schedule or on demand to create or update your assets' technical metadata. During the data source runs, you can optionally choose to publish your assets to the Amazon DataZone catalog and thus make them discoverable by all domain users. You can also publish your project inventory assets after editing their business metadata. Domain users can search for and discover your published assets, and request subscriptions to these assets.

To add an AWS Glue data source

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project to which you want to add the data source.
3. Navigate to the **Data** tab for the project.
4. Choose **Data sources** from the left navigation pane, then choose **Create data source**.
5. Configure the following fields:
   - **Name –** The data source name.
   - **Description –** The data source description.
6. Under **Data source type**, choose **AWS Glue**.
7. Under **Select an environment**, specify an environment in which to publish the AWS Glue tables.
8. Under **Data selection**, provide an AWS Glue database and enter your table selection criteria. For example, if you choose **Include** and enter `*corporate`, the database will include all source tables that end with the word `corporate`.

You can either choose an AWS Glue database form the dropdown or type a database name. The dropdown includes two databases: the publishing database and the subscription database of the environment. If you want to bring assets form a database that is not created by the environment, then you must type the name of the database instead of selecting it from the dropdown.

You can add multiple include and exclude rules for tables within a single database. You can also add multiple databases using the **Add another database** button.

9. Under **Data quality**, you can choose to **Enable data quality for this data source**. If you do this, Amazon DataZone imports your existing AWS Glue data quality output into your Amazon DataZone catalog. By default, Amazon DataZone imports the latest existing 100 quality reports with no expiration date from AWS Glue.

   Data quality metrics in Amazon DataZone help you understand the completeness and accuracy of your data sources. Amazon DataZone pulls these data quality metrics from AWS Glue in order to provide context during a point in time, for example, during a business data catalog search. Data users can see how data quality metrics change over time for their subscribed assets. Data producers can ingest AWS Glue data quality scores on a schedule. The Amazon DataZone business data catalog can also display data quality metrics from third-party systems through data quality APIs. For more information, see Data quality in Amazon DataZone
10. Choose **Next**.
11. For **Publishing settings**, choose whether assets are immediately discoverable in the business data catalog. If you only add them to the inventory, you can choose subscription terms later and publish them to the business data catalog. For more information, see the section called "Manage existing data sources".
12. For **Automated business name generation**, choose whether to automatically generate metadata for assets as they're imported from the source.
13. (Optional) For **Metadata forms**, add forms to define the metadata that is collected and saved when the assets are imported into Amazon DataZone. For more information, see the section called "Create, edit, or delete metadata forms".
14. For **Run preference**, choose when to run the data source.

    - **Run on a schedule** – Specify the dates and time to run the data source.
    - **Run on demand** – You can manually initiate data source runs.
15. Choose **Next**.

16. Review your data source configuration and choose **Create**.

> **Note**
>
> When an AWS Glue data source is created, Amazon DataZone creates the Lake Formation 'read only' permissions for the IAM role of the environment that is used to create the data source to access all the tables in the AWS Glue databases used in the data source. You can monitor the status of these grants under data sources on your environment's details page. Amazon DataZone adds the following AWS tags to the AWS Glue database when granting access to the publishing environment’s IAM role: `DataZoneDiscoverable_${domainId}: true`
> For the environments created prior to the current release of Amazon DataZone, project members will not be able to see granted tables in Amazon Athena.

Create and run an Amazon DataZone data source for Amazon Redshift

In Amazon DataZone, you can create an Amazon Redshift data source in order to import technical metadata of database tables and views from the Amazon Redshift data warehouse. To add a Amazon DataZone data source for Amazon Redshift, the source data warehouse must already exist in the Amazon Redshift.

When you create and run an Amazon Redshift data source, you add assets from the source Amazon Redshift data warehouse to your Amazon DataZone project's inventory. You can run your Amazon Redshift data sources on a set schedule or on demand to create or update your assets' technical metadata. During the data source runs, you can optionally choose to publish your project inventory assets to the Amazon DataZone catalog and thus make them discoverable by all domain users. You can also publish your inventory assets after editing their business metadata. Domain users can search for and discover your published assets and request subscriptions to these assets.

To add an Amazon Redshift data source

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.

2. Choose **Select project** from the top navigation pane and select the project to which you want to add the data source.
3. Navigate to the **Data** tab for the project.
4. Choose **Data sources** from the left navigation pane, then choose **Create data source.**
5. Configure the following fields:
   - **Name –** The data source name.
   - **Description –** The data source description.
6. Under **Data source type**, choose **Amazon Redshift**.
7. Under **Select an environment**, specify an environment in which to publish the Amazon Redshift tables.
8. Depending on the environment you select, Amazon DataZone will automatically apply the Amazon Redshift credentials and other parameters directly from the environment or give you the option to choose your own.
   - If you have selected an environment that only allows publishing from environment's default Amazon Redshift schema, then Amazon DataZone will automatically apply the Amazon Redshift credentials and other parameters including the Amazon Redshift cluster or workgroup name, AWS secret, database name, and schema name. You cannot edit these auto-populated parameters.
   - If you select an environment that does not allow to publish any data, you will not be able to proceed with data source creation.
   - If you select an environment that allows publishing data from any schema, you will see the option to either use the credentials and other Amazon Redshift parameters from the environment or to enter your own credentials/parameters.
9. If you choose to use your own credentials to create the data source, provide the following details:
   - Under **Provide Amazon Redshift credentials**, choose whether to use a provisioned Amazon Redshift cluster or an Amazon Redshift Serverless workspace as your data source.
   - Depending on your selection in the step above, choose your Amazon Redshift cluster or workspace from the dropdown menu, then choose the secret in AWS Secrets Manager to use for authentication. You can choose an existing secret or create a new one.
   - In order for the existing secret to appear in the drop down, make sure that your secret in AWS Secrets Manager includes the following tags (key/value):

- AmazonDataZoneProject: <projectID>
- AmazonDataZoneDomain: <domainID>

If you choose to create a new secret, then the secret is automatically tagged with the tags referenced above and no extra steps are needed. For more information, see Storing database credentials in AWS Secrets Manager.

Amazon Redshift users in the AWS secret provided for creating the data source must have `SELECT` permissions on the tables that are to be published. If you want Amazon DataZone to also manage the subscriptions (access) on your behalf, the database users in the AWS secret must also have the following permissions:

- `CREATE DATASHARE`
- `ALTER DATASHARE`
- `DROP DATASHARE`

10. Under **Data selection**, provide an Amazon Redshift database, schema, and enter your table or view selection criteria. For example, if you choose **Include** and enter `*corporate`, the asset will include all source tables that end with the word `corporate`.

    You can add multiple include rules for tables within a single database. You can also add multiple databases using the **Add another database** button.
11. Choose **Next**.
12. For **Publishing settings**, choose whether assets are immediately discoverable in the data catalog. If you only add them to the inventory, you can choose subscription terms later and publish them to the business data catalog. For more information, see the section called "Manage existing data sources".
13. For **Automated business name generation**, choose whether to automatically generate metadata for assets as they're published and updated from the source.
14. (Optional) For **Metadata forms**, add forms to define the metadata that is collected and saved when the assets are imported into Amazon DataZone. For more information, see the section called "Create, edit, or delete metadata forms".
15. For **Run preference**, choose when to run the data source.

    - **Run on a schedule** – Specify the dates and time to run the data source.
    - **Run on demand** – You can manually initiate data source runs.
16. Choose **Next**.

17. Review your data source configuration and choose **Create**.

> **Note**
>
> When an Amazon Redshift data source is created, Amazon DataZone grants read only' access to the environment used to create the data source to access all the tables in the Amazon Redshift schemas used in the data source. You can monitor the status of these grants under data sources on your environment's details page.
> When using a different Amazon Redshift cluster or a Serverless workgroup than the one used to create the environment, you must ensure that the following AWS tag is added to the cluster or workgroup. This is necessary for the environment users to be able to view the granted database in the Amazon Redshift Query Editor V2: `DataZoneDiscoverable_${domainId}: true`
> For the environments created prior to the current release of Amazon DataZone, project members will not be able to see granted tables in Amazon Redshift.

Manage existing Amazon DataZone data sources

After you create an Amazon DataZone data source, you can modify it at any time to change the source details or the data selection criteria. When you no longer need a data source, you can delete it.

To complete these steps, you must have the **AmazonDataZoneFullAccess** AWS managed policy attached. For more information, see the section called "AWS managed policies".

Topics

- Edit a data source
- Delete a data source

Edit a data source

You can edit an Amazon DataZone data source to modify its data selection settings, including adding, removing, or changing the table selection criteria. You can also add and remove databases. You can't change the data source type or the environment in which a data source is published.

To edit a data source

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project to which the data source belongs.
3. Navigate to the **Data** tab for the project.
4. Choose **Data sources** from the left navigation pane, then choose the data source that you want to modify.
5. Navigate to the **Data source definition** tab and choose **Edit**.
6. Make your changes to the data source definition. You can update the data source details and make changes to the data selection criteria.
7. When you're done making changes, choose **Save**.

Delete a data source

When you no longer need an Amazon DataZone data source, you can remove it permenantly. After you delete a data source, all assets that originated from that data source are still available in the catalog, and users can still subscribe to them. However, the assets will stop receiving updates from the source. We recommend that you first move the dependent assets to a different data source before you delete it.

> **Note**
>
> You must remove all fulfillments on the data source before you can delete it. For more information, see *Discovering, subscribing to, and consuming data in Amazon DataZone*.

To delete a data source

1. On the **Data** tab for the project, choose **Data sources** from the left navigation pane.
2. Choose the data source that you want to delete.
3. Choose **Actions**, **Delete data source** and confirm deletion.

Publish assets to the Amazon DataZone catalog from the project inventory

You can publish Amazon DataZone assets and their metadata from project inventories into the Amazon DataZone catalog. You can only publish the most recent version of an asset to the catalog.

Consider the following when publishing assets to the catalog:

- To publish an asset to the catalog, you must be the owner or the contributor of that project.
- For Amazon Redshift assets, ensure that the Amazon Redshift clusters associated with both publisher and subscriber clusters meet all the requirements for Amazon Redshift data sharing in order for Amazon DataZone to manage access for Redshift tables and views. See Data sharing concepts for Amazon Redshift.
- Amazon DataZone only supports access management for assets published from the AWS Glue Data Catalog and Amazon Redshift. For all other assets, such as Amazon S3 objects, Amazon DataZone does not manage access for approved subscribers. If you subscribe to these unmanaged assets, you're notified with the following message:

  ```
  Subscription approval does not provide access to data. Subscription
  grants on this asset are not managed by Amazon DataZone. For more
  information or help, reach out to your administrator.
  ```

Publish an asset

If you didn't choose to make assets immediately discoverable in the data catalog when you created a data source, perform the following steps to publish them later.

To publish an asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Navigate to the **Data** tab for the project.

4. Choose **Inventory data** from the left navigation pane, then select the asset that you want to publish.

> **Note**
>
> By default, all assets require subscription approval, which means a data owner must approve all subscription requests to the asset. If you want to change this setting before publishing the asset, open the asset details and choose **Edit** next to **Subscription approval**. You can change this setting later by modifying and re-publishing the asset.

5. Choose **Publish asset**. The asset is directly published to the catalog.

   If you make changes to the asset, such as modifying its approval requirements , you can choose **Re-publish** to publish the updates to the catalog.

Manage inventory and curate assets

In order to use Amazon DataZone to catalog your data, you must first bring your data (assets) as inventory of your project in Amazon DataZone. Creating inventory for a particular project, makes the assets discoverable only to that project's members.

Once the assets are created in project inventory, their metadata can be curated. For example, you can edit the asset's name, description, or read me. Each edit to the asset creates a new version of the asset. You can use the History tab on the asset's details page to view all asset versions.

You can edit the **Read Me** section and add rich descriptions for the asset. The **Read Me** section supports markdown, thus enabling you to format your descriptions as required and describe key information about an asset to consumers.

Glossary terms can be added at the asset level by filling out available forms.

To curate the schema, you can review the columns, add business names, descriptions, and add glossary terms at column level.

If automated metadata generation is enabled when the data source is created, the business names for assets and columns are available to review and accept or reject individually or all at once.

You can also edit the subscription terms to specify if approval for the asset is required or not.

Metadata forms in Amazon DataZone enable you to extend a data asset's metadata model by adding custom-defined attributes (for example, sales region, sales year, and sales quarter). The metadata forms that are attached to an asset type are applied to all assets created from that asset type. You can also add additional metadata forms to individual assets as part of the data source run or after it's created. For creating new forms, see the section called “Create, edit, or delete metadata forms”.

To update the metadata of an asset, you must be the owner or the contributor of the project to which the asset belongs.

To update the metadata of an asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project that contains the asset whose metadata you want to update.
3. Navigate to the **Data** tab for the project.
4. Choose **Inventory data** from the left navigation pane, then choose the name of the the asset whose metadata you want to update.
5. On the asset details page, under **Metadata forms**, choose **Edit** and edit the existing forms as needed. You can also attach additional metadata forms to the asset. For more information, see the section called “Attach additional metadata forms to assets”.
6. When you're done making updates, choose **Save form**.

   When you save the form, Amazon DataZone generates a new inventory version of the asset. To publish the updated version to the catalog, choose **Re-publish asset**.

Attach additional metadata forms to assets

By default, metadata forms attached to a domain are attached to all assets published to that domain. Data publishers can associate additional metadata forms to individual assets in order to provide additional context.

To attach additional metadata forms to an asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project that contains the asset whose metadata you want to add to.
3. Navigate to the **Data** tab for the project.
4. Choose **Inventory data** from the left navigation pane, then choose the name of the the asset whose metadata you want to add to.
5. On the asset details page, under **Metadata forms**, choose **Add forms**.
6. Select the form(s) to add to the asset, then choose **Add forms**.
7. Enter values for each of the metadata fields, then choose **Save form**.

   When you save the form, Amazon DataZone generates a new inventory version of the asset. To publish the updated version to the catalog, choose **Re-publish asset**.

Publish asset to the catalog after curation

Once satisfied with the asset curation, the data owner can publish an asset version to the Amazon DataZone catalog and thus make it discoverable by all domain users. The asset shows the inventory version and the published version. In the discovery catalog, only the latest published version appears. If the metadata is updated after publishing, then a new inventory version will be available for publishing to the catalog.

Manually create an asset

In Amazon DataZone, an asset is an entity that presents a single physical data object (for example, a table, a dashboard, a file) or virtual data object (for example, a view). For more information, see Amazon DataZone terminology and concepts. Publishing an asset manually is a one-time operation. You don't specify a run schedule for the asset, so it's not updated automatically if its source changes.

To manually create an asset through a project, you must be the owner or the contributor of that project.

To create an asset manually

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project to which to create the asset.
3. Navigate to the **Data** tab for the project.
4. Choose **Data sources** from the left navigation pane, then choose **Create data asset**.
5. For **Asset details**, configure the following settings:

   - **Asset type** – The type of asset.
   - **Name** – The name of the asset.
   - **Description** – A description of the asset.
6. For **S3 location**, enter the Amazon Resource Name (ARN) of the source S3 bucket.

   Optionally, enter an S3 access point. For more information, see Managing data access with Amazon S3 access points.
7. For **Publishing settings**, choose whether assets are immediately discoverable in the catalog. If you only add them to the inventory, you can choose subscription terms later to publish them to the catalog.
8. Choose **Create**.

   Once the asset is created, it will either be directly published as an active asset in the catalog, or will be stored in the inventory until you decide to publish it.

Unpublish an asset from the Amazon DataZone catalog

When you unpublish an Amazon DataZone asset from the catalog, it no longer appears in global search results. New users won't be able to find or subscribe to the asset listing in the catalog, but all existing subscriptions remain the same.

To unpublish an asset, you must be the owner or the contributor of the project to which the asset belongs:

To unpublish an asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project to which the asset belongs.
3. Navigate to the **Data** tab for the project.
4. Choose **Published data** from the left navigation pane.
5. Locate the asset from the list of published assets, then choose **Unpublish**.

   The asset is removed from the catalog. You can re-publish the asset at any time by choosing **Publish**.

Delete an Amazon DataZone asset

When you no longer need an asset in Amazon DataZone, you can permanently delete it. Deleting an asset is different than unpublishing an asset from the catalog. You can delete an asset and its related listing in the catalog so that it's not visible in any search results. To delete the asset listing, you must first revoke all of its subscriptions.

To delete an asset, you must be the owner or the contributor of the project to which the asset belongs:

> **Note**
>
> In order to delete an asset listing, you must first revoke all existing subscriptions to the asset. You can't delete an asset listing that has existing subscribers.

To delete and asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.

2. Choose **Select project** from the top navigation pane and select the project that contains the asset that you want to delete.
3. Navigate to the **Data** tab for the project.
4. Choose **Published data** from the left navigation pane, then locate and choose the asset that you want to delete. This opens the asset details page.
5. Choose **Actions**, **Delete** and confirm deletion.

   Once the asset is deleted, it's no longer available to view and users can't subscribe to it.

Manually start a data source run in Amazon DataZone

When you run a data source, Amazon DataZone pulls all any new or modified metadata from the source and updates the associated assets in the inventory. When you add a data source to Amazon DataZone, you specify the source's run preference, which defines whether the source runs on a schedule or on demand. If your source runs on demand, you must initiate a data source run manually.

Even if your source runs on a schedule, you can still run it manually at any time. After adding business metadata to the assets, you can select assets and publish them to the Amazon DataZone catalog in order for these assets to be discoverable by all domain users. Only published assets are searchable by other domain users.

To run a data source manually

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project to which the data source belongs.
3. Navigate to the **Data** tab for the project.
4. Choose **Data sources** from the left navigation pane, then locate and choose the data source that you want to run. This opens the data source details page.
5. Choose **Run on demand**.

The data source status changes to `Running` as Amazon DataZone updates the asset metadata with the most recent data from the source. You can monitor the status of the run on the **Data source runs** tab.

Asset revisions in Amazon DataZone

Amazon DataZone increments the revision of an asset when you edit its business or technical metadata. These edits include modifying the asset name, description, glossary terms, columns names, metadata forms, and metadata form field values. These changes can result from manual edits, data source job run, or API operations. Amazon DataZone automatically generates a new asset revision any time you make an edit to the asset.

After you update an asset and a new revision is generated, you must publish the new revision to the catalog for it to be updated and available to subscribers. For more information, see the section called "Publish assets to the catalog from the project inventory". You can only publish the most recent version of an asset to the catalog.

To view past revisions of an asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project that contains the asset.
3. Navigate to the **Data** tab for the project, then locate and choose the asset. This opens the asset details page.
4. Navigate to the **History** tab, which displays a list of past revisions of the asset.

Data quality in Amazon DataZone

Data quality metrics in Amazon DataZone help you understand the different quality metrics such as completeness, timeliness, and accuracy of your data sources. Amazon DataZone integrates with AWS Glue Data Quality and offers APIs to integrate data quality metrics from third-party data quality solutions. Data users can see how data quality metrics change over time for their subscribed assets. To author and run the data quality rules, you can use your data quality tool of choice such as AWS Glue data quality. With data quality metrics in Amazon DataZone, data consumers can visualize the data quality scores for the assets and columns, helping build trust in the data they use for decisions.

Pre-requisites and IAM role changes

If you are using Amazon DataZone's AWS managed policies, there are no additional configuration steps and these managed policies are automatically updated to support data quality. If you are using your own policies for the roles that grant Amazon DataZone the required permissions to interoperate with supported services, you must update the policies attached to these roles to enable support for reading the AWS Glue data quality information in the AWS managed policy: AmazonDataZoneGlueManageAccessRolePolicy and enable support for the time series APIs in the AWS managed policy: AmazonDataZoneDomainExecutionRolePolicy and the AWS managed policy: AmazonDataZoneFullUserAccess.

Enabling data quality for AWS Glue assets

Amazon DataZone pulls the data quality metrics from AWS Glue in order to provide context during a point in time, for example, during a business data catalog search. Data users can see how data quality metrics change over time for their subscribed assets. Data producers can ingest AWS Glue data quality scores on a schedule. The Amazon DataZone business data catalog can also display data quality metrics from third-party systems through data quality APIs. For more information, see AWS Glue Data Quality and Getting started with AWS Glue Data Quality for the Data Catalog.

You can enable data quality metrics for your Amazon DataZone assets in the following ways:

- Use the Data Portal or the Amazon DataZone APIs to enable data quality for your AWS Glue data source via the Amazon DataZone data portal either while creating new or editing existing AWS Glue data source.

  For more information on enabling data quality for a data source via the portal, see Create and run an Amazon DataZone data source for the AWS Glue Data Catalog and Manage existing Amazon DataZone data sources.

  > **Note**
  >
  > You can use the Data Portal to enable data quality only for your AWS Glue inventory assets. In this release of Amazon DataZone enabling data quality for Amazon Redshift or custom types assets via the data portal is not supported.

You can also use the APIs to enable data quality for your new or existing data sources. You can do this by invoking the [CreateDataSource]() or [UpdateDataSource]() and setting the `autoImportDataQualityResult` parameter to 'True'.

After data quality is enabled, you can run the data source on demand or on schedule. Each run can bring in up to 100 metrics per asset. There is no need to create forms or add metrics manually when using data source for data quality. When the asset is published, the updates that were made to the data quality form (up to 30 data points per rule of history) are reflected in the listing for the consumers. Subsequently, each new addition of metrics to the asset, is automatically added to the listing. There is no need to republish the asset to make the latest scores available to consumers.

Enabling data quality for custom asset types

You can use the Amazon DataZone APIs to enable data quality for any of your custom type assets. For more information, see the following:

- PostTimeSeriesDataPoints
- ListTimeSeriesDataPoints
- GetTimeSeriesDataPoint
- DeleteTimeSeriesDataPoints

The following steps provide an example of using APIs or CLI to import third-party metrics for your assets in Amazon DataZone:

1. Invoke the `PostTimeSeriesDataPoints` API as follows:

   ```
   aws datazone post-time-series-data-points  \
   --cli-input-json file://createTimeSeriesPayload.json \
   ```

   with the following payload:

   ```
   "domainId": "dzd_5oo7xzoqltu8mf",
   ```

```
    "entityId": "4wyh64k2n8czaf",
    "entityType": "ASSET",
    "form": {
        "content": "{\n  \"evaluations\" : [ {\n    \"types\" : [ \"MaximumLength
\" ],\n    \"description\" : \"ColumnLength \\\"ShippingCountry\\\" <= 6\",\n
   \"details\" : { },\n    \"applicableFields\" : [ \"ShippingCountry\" ],\n
   \"status\" : \"PASS\"\n  }, {\n    \"types\" : [ \"MaximumLength\" ],\n
 \"description\" : \"ColumnLength \\\"ShippingState\\\" <= 2\",\n    \"details
\" : { },\n    \"applicableFields\" : [ \"ShippingState\" ],\n    \"status\" :
 \"PASS\"\n  }, {\n    \"types\" : [ \"MaximumLength\" ],\n    \"description
\" : \"ColumnLength \\\"ShippingCity\\\" <= 8\",\n    \"details\" : { },\n
 \"applicableFields\" : [ \"ShippingCity\" ],\n    \"status\" : \"PASS\"\n  },
 {\n    \"types\" : [ \"Completeness\" ],\n    \"description\" : \"Completeness \
\\"ShippingStreet\\\" >= 0.59\",\n    \"details\" : { },\n    \"applicableFields
\" : [ \"ShippingStreet\" ],\n    \"status\" : \"PASS\"\n  }, {\n    \"types\" :
 [ \"MaximumLength\" ],\n    \"description\" : \"ColumnLength \\\"ShippingStreet\\
\" <= 101\",\n    \"details\" : { },\n    \"applicableFields\" : [ \"ShippingStreet
\" ],\n    \"status\" : \"PASS\"\n  }, {\n    \"types\" : [ \"MaximumLength\" ],\n
    \"description\" : \"ColumnLength \\\"BillingCountry\\\" <= 6\",\n    \"details
\" : { },\n    \"applicableFields\" : [ \"BillingCountry\" ],\n    \"status\" :
 \"PASS\"\n  }, {\n    \"types\" : [ \"Completeness\" ],\n    \"description\" :
 \"Completeness \\\"biLlingcountry\\\" >= 0.5\",\n    \"details\" : {\n
 \"EVALUATION_MESSAGE\" : \"Value: 0.26666666666666666 does not meet the constraint
 requirement!\"\n    },\n    \"applicableFields\" : [ \"biLlingcountry\" ],\n
    \"status\" : \"FAIL\"\n  }, {\n    \"types\" : [ \"Completeness\" ],\n
 \"description\" : \"Completeness \\\"Billingstreet\\\" >= 0.5\",\n    \"details
\" : { },\n    \"applicableFields\" : [ \"Billingstreet\" ],\n    \"status\" :
 \"PASS\"\n  } ],\n  \"passingPercentage\" : 88.0,\n  \"evaluationsCount\" : 8\n}",
        "formName": "shortschemaruleset",
        "id": "athp9dyw75gzhj",
        "timestamp": 1.71700477757E9,
        "typeIdentifier": "amazon.datazone.DataQualityResultFormType",
        "typeRevision": "8"
    },
    "formName": "shortschemaruleset"
}
```

You can obtain this payload by invoking the `GetFormType` action:

```
aws datazone get-form-type --domain-identifier <your_domain_id> --form-type-
identifier amazon.datazone.DataQualityResultFormType --region <domain_region> --
output text --query 'model.smithy'
```

2. Invoke the `DeleteTimeSeriesDataPoints` API as follows:

```
aws datazone delete-time-series-data-points\
--domain-identifier dzd_bqqlk3nz21zp2f \
--entity-identifier dzd_bqqlk3nz21zp2f \
--entity-type ASSET \
--form-name rulesET1 \
```

Using machine learning and generative AI

> **Note**
>
> Powered by Amazon Bedrock: AWS implements automated abuse detection. Because the AI recommendations for descriptions functionality in Amazon DataZone is built on Amazon Bedrock, users inherit the controls implemented in Amazon Bedrock to enforce safety, security, and the responsible use of AI.

In the current release of Amazon DataZone, you can use the AI recommendations for descriptions functionality to automate data discovery and cataloging. Support for generative AI and machine learning in Amazon DataZone creates descriptions for assets and columns. You can use these descriptions to add business context for your data and recommend analysis for datasets, which can help boost data discovery results.

Powered by Amazon Bedrock's large language models, the AI recommendations for data asset descriptions in Amazon DataZone help you to ensure that your data is comprehensible and easily discoverable. The AI recommendations also suggest the most pertinent analytical applications for datasets. By reducing manual documentation tasks and advising on appropriate data usage, auto-generated descriptions can help you to enhance the trustworthiness of your data and minimize overlooking valuable data to accelerate informed decision making.

⚠ **Important**

In the current Amazon DataZone release, the AI recommendations for descriptions feature is only supported in the following regions:

- US East (N. Virginia)
- US West (Oregon)
- Europe (Frankfurt)
- Asia Pacific (Tokyo)

The following procedure describes how to generate AI recommendations for descriptions in Amazon DataZone:

1. Navigate to the Amazon DataZone data portal URL, and then sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, and then choose **Open data portal**.
2. In the top navigation pane, choose **Select project**, and then choose the project that contains the asset for which you want to generate AI recommendations for descriptions.
3. Navigate to the **Data** tab for the project.
4. In the left navigation pane, choose **Inventory data**, and then choose the name of the asset for which you want to generate AI recommendations for descriptions for the asset.
5. On the asset's details page, in the **Business metadata** tab, choose **Generate descriptions**.
6. Once the descriptions are generated, you can either edit, accept, or reject them. Green icons are displayed next to each automatically generated metadata description for the data asset. In the **Business metadata** tab, you can choose the green icon next to the automatically generated **Summary**, and then choose **Edit**, **Accept**, or **Reject** to address the generated description. You can also choose **Accept all** or **Reject all** options that are displayed at the top of the page when the **Business metadata** tab is selected, and thus perform the selected action on all automatically generated descriptions.

   Or you can choose the **Schema** tab, and then address automatically generated descriptions individually by choosing the green icon for one column description at a time and then choosing **Accept** or **Reject**. In the **Schema** tab, you can also choose **Accept all** or **Reject all** and thus perform the selected action on all automatically generated descriptions.

7. To publish the asset to the catalog with the generated descriptions, choose **Publish asset**, and then confirm this action by choosing **Publish asset** again in the **Publish asset** pop up window.

   > **Note**
   >
   > If you don't accept or reject the generated descriptions for an asset, and then you publish this asset, this unreviewed automatically generated metadata is not included in the published data asset.

Data lineage in Amazon DataZone (Preview)

> **Important**
>
> Currently, the data lineage functionality in Amazon DataZone is in a Preview release.

Data lineage in Amazon DataZone is an API-driven, OpenLineage-compatible feature that can help you to capture and visualize lineage events, from OpenLineage-enabled systems or through APIs, to trace data origins, track transformations, and view cross-organizational data consumption. It provides you with an overarching view into your data assets to see the origin of assets and their chain of connections. The lineage data includes information on the activities inside the Amazon DataZone's business data catalog, including information about the catalogued assets, the subscribers of those assets, and the activities that happen outside the business data catalog captured programmatically using the APIs.

Using Amazon DataZone's OpenLineage-compatible APIs, domain administrators and data producers can capture and store lineage events beyond what is available in Amazon DataZone, including transformations in Amazon S3, AWS Glue, and other services. This provides a comprehensive view for the data consumers and helps them gain confidence of the asset's origin, while data producers can assess the impact of changes to an asset by understanding its usage. Additionally, Amazon DataZone versions lineage with each event, enabling users to visualize lineage at any point in time or compare transformations across an asset's or job's history. This historical lineage provides a deeper understanding of how data has evolved, essential for troubleshooting, auditing, and ensuring the integrity of data assets.

With data lineage, you can accomplish the following in Amazon DataZone:

- Understand the provenance of data: knowing where the data originated fosters trust in data by providing you with a clear understanding of its origins, dependencies, and transformations. This transparency helps in making confident data-driven decisions.
- Understand the impact of changes to data pipelines: when changes are made to data pipelines, lineage can be used to identify all of the downstream consumers that are to be affected. This helps to ensure that changes are made without disrupting critical data flows.
- Identify the root cause of data quality issues: if a data quality issue is detected in a downstream report, lineage, especially column-level lineage, can be used to trace the data back (at a column level) to identify the issue back to its source. This can help data engineers to identify and fix the problem.
- Improve data governance and compliance: column-level lineage can be used to demonstrate compliance with data governance and privacy regulations. For example, column-level lineage can be used to show where sensitive data (such as PII) is stored and how it is processed in downstream activities.

Types of lineage nodes in Amazon DataZone

in Amazon DataZone, data lineage information is presented in nodes that represent tables and views. Depending on the context of the project, for example, a project selected at the top left in the data portal, the producers are able to view both, inventory and published assets, whereas consumers can only view the published assets. When you first open the lineage tab in the asset details page, the catalogued dataset node is the starting point for navigating upstream or downstream through the lineage nodes of your lineage graph.

The following are the types of data lineage nodes that are supported in Amazon DataZone:

- **Dataset node** - this node type includes data lineage information about a specific data asset.
  - Dataset nodes that include information about AWS Glue or Amazon Redshift assets published in the Amazon DataZone catalog are auto-generated and include a corresponding AWS Glue or Amazon Redshift icon within the node.
  - Dataset nodes that include information about assets that are not published in the Amazon DataZone catalog, are created manually by domain administrators (producers) and are represented by a default custom asset icon within the node.
- **Job (run) node** - this node type displays the details of the job, including the latest run of a particular job and run details. This node also captures multiple runs of the job and can be viewed in the **History** tab of the node details. You can view node details by choosing the node icon.

Key attributes in lineage nodes

The `sourceIdentifier` attribute in a lineage node represents the events happening on a dataset. The `sourceIdentifier` of the lineage node is the identifier of the dataset (table/view etc). It's used for uniqueness enforcement on the lineage nodes. For example, there can't be two lineage nodes with same `sourceIdentifier`. The following are examples of `sourceIdentifier` values for different types of nodes:

- For dataset node with respective dataset type:
  - Asset: amazon.datazone.asset/<assetId>
  - Listing (published asset): amazon.datazone.listing/<listingId>
  - AWS Glue table: arn:aws:glue:<region>:<account-id>:table/<database>/<table-name>
  - Amazon Redshift table/view: arn:aws:<redshift/redshift-serverless>:<region>:<account-id>:<table-type(table/view etc)>/<clusterIdentifier/workgroupName>/<database>/<schema>/<table-name>
  - For any other type of dataset nodes imported using open-lineage run events, <namespace>/<name> of the input/output dataset is used as `sourceIdentifier` of the node.
- For jobs:
  - For job nodes imported using open-lineage run events, <jobs_namespace>.<job_name> is used as sourceIdentifier.
- For job runs:
  - For job run nodes imported using open-lineage run events, <jobs_namespace>.<job_name>/<run_id> is used as sourceIdentifier.

For assets created using `createAsset` API, the `sourceIdentifier` must be updated using `createAssetRevision` API to enable mapping the asset to upstream resources.

Visualizing data lineage

Amazon DataZone's asset details page provides a graphical representation of data lineage, making it easier to visualize data relationships upstream or downstream. The asset details page provides the following capabilities to navigate the graph:

- Column-level lineage: expand column-level lineage when available in dataset nodes. This automatically shows relationships with upstream or downstream dataset nodes if source column information is available.

- Column search: when the default display for number of columns is 10. If there are more than 10 columns, pagination is activiated to navigate to the rest of the columns. To quickly view a particular column, you can search on the dataset node that list just the searched column.
- View dataset nodes only: if you want to toggle to view only dataset lineage nodes and filter out the job nodes, you can choose the Open view control icon on the top left of the graph viewer and toggle the **Display dataset nodes only** option. This will remove all the job nodes from the graph and lets you navigate just the dataset nodes. Note that when the view only dataset nodes is turned on, the graph cannot be expanded upstream or downstream.
- Details pane: Each lineage node has details captured and displayed when selected.
  - Dataset node has a detail pane to display all the details captured for that node for a given timestamp. Every dataset node has 3 tabs, namely: Lineage info, Schema, and History tab. The history tab lists the different versions of lineage event captured for that node. All details captured from API are displayed using metadata forms or a JSON viewer.
  - Job node has a detail pane to display job details with tabs, namely: Job info, and History. The details pane also captures query or expressions captured as part of the job run. The history tab lists the different versions of job run event captured for that job. All details captured from API are displayed using metadata forms or a JSON viewer.
- Version tabs: all lineage nodes in Amazon DataZone data lineage have versioning. For every dataset node or job node, the versions are captured as history and that enables you to navigate between the different versions to identify what has changed overtime. Each version opens a new tab in the lineage page to help compare or contrast.

Data lineage authorization in Amazon DataZone

Write permissions - to publish lineage data into Amazon DataZone, you must have an IAM role with a permissions policy that includes an `ALLOW` action on the `PostLineageEvent` API. This IAM authorization happens at API Gateway layer.

Read permissions - there are two operations: `GetLineageNode` and `ListLineageNodeHistory` that are included in the `AmazonDataZoneDomainExecutionRolePolicy` managed policy and therefore every user in the Amazon DataZone domain can invoke these to traverse the data lineage graph.

Data lineage sample experience in Amazon DataZone

You can use the data lineage sample experience to browse and understand data lineage in Amazon DataZone, including traversing upstream or downstream in your data lineage graph, exploring versions and column-level lineage.

Complete the followng procedure to try the sample data lineage experience in Amazon DataZone:

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose any available data asset to open the asset's details page.
3. On the asset's details page, choose the **Lineage** tab and then choose **Preview**, and then choose **Try sample lineage** .
4. In the data lineage pop up window, choose **Start guided data lineage tour**.

   At this point, a fullscreen tab that provides all the space of lineage information is displayed. The sample data lineage graph is initially displayed with a base node with 1-depth at either ends, upstream and downstream. You can expand the graph upstream or downstream. The columns information is also available for you to choose and see how lineage flows through the nodes.

Using Amazon DataZone data lineage programmatically

To use the data lineage functionality in Amazon DataZone, you can invoke the following APIs:

- GetLineageNode
- ListLineageNodeHistory
- PostLineageEvent

Discovering, subscribing to, and consuming data in Amazon DataZone

In Amazon DataZone, once an asset is published to a domain, subscribers can discover and request a subscription to this asset. The subscription process begins with a subscriber searching for and browsing the catalog to find an asset they want. From the Amazon DataZone portal, they choose to subscribe to the asset by submitting a subscription request that includes justification and the reason for the request. The subscription approver, as defined in the publishing agreement, then reviews the access request. They can either approve or reject the request.

After a subscription is granted, a fulfillment process starts to facilitate access to the asset for the subscriber. There are two primary modes of asset access control and fulfillment: those for Amazon DataZone-managed assets and those for assets that are not managed by Amazon DataZone.

- **Managed assets** – Amazon DataZone can manage fulfillment and permissions for managed assets, such as AWS Glue tables and Amazon Redshift tables and views.
- **Unmanaged assets** – Amazon DataZone publishes standard events related to your actions (for example, approval given to a subscription request) to Amazon EventBridge. You can use these standard events to integrate with other AWS services or third-party solutions for custom integrations.

Topics

Discovering data

The following tasks describe various ways to discover data in Amazon DataZone.

Topics

Search for and view assets in the catalog

Amazon DataZone provides a streamlined way to search for data. Any Amazon DataZone user with permissions to access the data portal can search for assets in the Amazon DataZone catalog and view asset names and the metadata assigned to them. You can take a closer look at an asset by examining its details page.

> **Note**
>
> To view the actual data that an asset contains, you must first subscribe to the asset and have your subscription request approved and access granted. For more information, see Subscribing to data.

To search for assets in the catalog

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. You can type the name of the asset that you are looking for in the search bar on the home page of the data portal.
3. To browse namespaces, choose **Catalog** from the top right of the page to open the catalog. The catalog provides a faceted search experience for you to find assets by searching on criteria such as , data owner, and glossary terms.
4. Enter your search term in one of the search boxes. After you run a search, you can apply various filters to narrow the results. The filters include asset type, source account, and the AWS Region to which the asset belongs.
5. To view details about a specific asset, choose the asset to open its details page. The details page includes the following information:
   - The asset name, data source (AWS Glue, Amazon Redshift, or Amazon S3), type (table, view, or S3 object), number of columns, and size.
   - A description of the asset.
   - The current published revision of the asset, the owner, whether approval is required for subscriptions, the namepace, and update history.

- An **Overview** tab which includes glossary terms and metadata forms.
- A **Schema** tab which displays the schema of the asset, including business and technical column names, data types, and business descriptions of the columns. The schema tab is visible only for tables and views (not for Amazon S3 objects).
- A **Subscriptions** tab which includes a list of subscribers to the domain.
- A **History** tab which includes a list of past revisions of the asset.

Subscribing to data

The following tasks provide details on subscribing to assets in Amazon DataZone.

Topics

Request subscription to assets

Amazon DataZone allows you to find, access and consume the assets in the Amazon DataZone catalog. When you find an asset in the catalog that you want to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request.

You must be a member of a project in order to request subscription to an asset within that project.

To subscribe to an asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Use the search bar to search for and choose the asset to which you want to subscribe, and then choose **Subscribe**.

3. In the **Subscribe** pop up window, provide the following information:

   - The project that you want to subscribe to the asset.
   - A short justification for your subscription request.

4. Choose **Subscribe**.

   You receive a notification in the data portal when the publisher approves your request.

To view the status of the subscription request, locate and choose the project with which you subscribed to the asset. Navigate to the **Data** tab for the project, then choose **Requested data** from the left navigation pane. This page lists the assets to which the project has requested access. You can filter the list by the status of the request.

Approve or reject a subscription request

Amazon DataZone allows you to find, access and consume the assets in the Amazon DataZone catalog. When you find an asset in the catalog that you want to access, you must *subscribe* to the asset, which creates a subscription request. An approver can then approve or reject your request.

You must be a member of the owning project (the project that published the asset) to approve or reject a subscription request.

To approve or reject a subscription request

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. In the data portal, choose **Browse projects list** and select the project that contains the asset with the subscription request.
3. Navigate to the **Data** tab, then choose **Incoming requests** from the left navigation pane.
4. Locate the request and choose **View request**. You can filter by **Pending** to see only requests that are still open.
5. Review the subscription request and reason for access, and decide whether to approve or reject it.
6. (Optional) Enter a response that explains your reason for accepting or rejecting the request.
7. Choose either **Approve** or **Reject**.

As the project owner, you can revoke the subscription at any time. For more information, see the section called “Revoke an existing subscription”.

To view all subscription requests, see *Working with Amazon DataZone events and notifications*.

Revoke an existing subscription

Amazon DataZone allows you to find, access and consume the assets in the Amazon DataZone catalog. When you find an asset in the catalog that you want to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request. You might need to revoke a subscription after you have approved it, either because the approval was a mistake, or because the subscriber no longer needs access to the asset.

You must be a member of the owning project (the project that published the asset) to revoke a subscription.

To revoke a subscription

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project that contains the subscription you want to revoke.
3. Navigate to the **Data** tab, then choose **Incoming requests** from the left navigation pane.
4. Locate the subscription you want to revoke and choose **View subscription**.
5. (Optional) Enable the checkbox to allow the subscriber to keep the asset in the project's subscription targets. A subscription target is a reference to a set of resources where subscribed data can be made available within an environment.

   If you want to revoke access to the asset from the subscription target at a later time, you must do so in AWS Lake Formation.
6. Choose **Revoke subscription**.

You can't re-approve a subscription after you revoke it. The subscriber must subscribe to the asset again in order for you to approve it.

Cancel a subscription request

Amazon DataZone allows you to find, access and consume the assets in the Amazon DataZone catalog. When you find an asset in the catalog that you want to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request. You might need to cancel a pending subscription request, either because you submitted it by mistake, or because you no longer need read access to the asset.

To cancel a subscription request, you must be either a project owner or contributor.

To cancel a subscription request

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you're an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project that contains the subscription request.
3. Navigate to the **Data** tab for the project, then choose **Requested data** from the left navigation pane. This page lists the assets to which the project has requested access.
4. Filter by **Requested** to see only requests that are still pending. Locate the request and choose **View request**.
5. Review the subscription request and choose **Cancel request**.

If you want to re-subscribe to the asset (or to a different asset), see the section called "Request subscription to assets".

Unsubscribe from an asset

Amazon DataZone allows you to find, access and consume the assets in the Amazon DataZone catalog. When you find an asset in the catalog that you want to access, you need to *subscribe* to the asset, which creates a subscription request. An approver can then approve or request your request. You might need to unsubscribe from an asset, either because you subscribed by mistake and were approved, or because you no longer need read access to the asset.

You must be a member of a project in order to unsubscribe from one of its assets.

To unsubscribe from an asset

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.
2. Choose **Select project** from the top navigation pane and select the project that contains the asset you want to unsubscribe from.
3. Navigate to the **Data** tab for the project, then choose **Requested data** from the left navigation pane. This page lists the assets to which the project has requested access.
4. Filter by **Approved** to see only requests that have been approved. Locate the request and choose **View subscription**.
5. Review the subscription and choose **Unsubscribe**.

If you want to re-subscribe to the asset (or to a different asset), see the section called “Request subscription to assets”.

Using existing IAM roles to fulfill Amazon DataZone subscriptions

In the current release, Amazon DataZone supports you using your existing IAM roles to get access to the data. To achieve this, you can create a subscription target in the Amazon DataZone environment that you're using to fulfill your subscription. To create a subscription target for an environment in one of the associated AWS accounts, you can use the following steps:

Step 1: Ensure that your Amazon DataZone domain is using version 2 or higher of the RAM policy

1. Navigate to the **Shared by me : Resource shares** page in the AWS RAM console.
2. Because AWS RAM resource shares exist in specific AWS Regions, choose the appropriate AWS Region from the dropdown list in the upper-right corner of the console.
3. Select the resource share corresponding to your Amazon DataZone domain and then choose **Modify**. You can identify the RAM share for the Amazon DataZone domain using the name or ID of the domain as the RAM share is created with the name: `DataZone-<domain-name>-<domain-id>`.
4. Choose **Next** to proceed to the next step where you can check the version of the RAM policy and modify it.

5. Make sure that the version of the RAM policy is Version 2 or higher. If not, use the dropdown to select Version 2 or higher.
6. Choose **Skip to step 4: Review and update**.
7. Choose **Update resource share**.

Step 2: Create a subscription target from an associated account

- In the current release, Amazon DataZone supports creating subscription targets by using APIs only. Below are some examples of the payload you can use to create a subscription target for fulfilling subscriptions to your AWS Glue tables and Amazon Redshift tables or views. For more information, see CreateSubscriptionTarget.

  Example of subscription target for AWS Glue

  ```
  {
          "domainIdentifier": "<DOMAIN_ID>",
          "environmentIdentifier": "<ENVIRONMENT_ID>",
          "name": "<SUBSCRIPTION_TARGET_NAME>",
          "type": "GlueSubscriptionTargetType",
          "authorizedPrincipals" : ["IAM_ROLE_ARN"],
          "subscriptionTargetConfig" : [{"content": "{\"databaseName\":
   \"<DATABASE_NAME>\"}", "formName": "GlueSubscriptionTargetConfigForm"}],
          "manageAccessRole": "<GLUE_DATA_ACCESS_ROLE_IN_ASSOCIATED_ACCOUNT_ARN>",
          "applicableAssetTypes" : ["GlueTableAssetType"],
          "provider": "Amazon DataZone"
  }
  ```

  Example of subscription target for Amazon Redshift:

  ```
  {
          "domainIdentifier": "<DOMAIN_ID>",
          "environmentIdentifier": "<ENVIRONMENT_ID>",
          "name": "<SUBSCRIPTION_TARGET_NAME>",
          "type": "RedshiftSubscriptionTargetType",
          "authorizedPrincipals" : ["REDSHIFT_DATABASE_ROLE_NAME"],
          "subscriptionTargetConfig" : [{"content": "{\"databaseName\":
   \"<DATABASE_NAME>\", \"secretManagerArn\": \"<SECRET_MANAGER_ARN>
  ```

```
\",\"clusterIdentifier\": \"<CLUSTER_IDENTIFIER>\"}", "formName":
 "RedshiftSubscriptionTargetConfigForm"}],
        "manageAccessRole":
 "<REDSHIFT_DATA_ACCESS_ROLE_IN_ASSOCIATED_ACCOUNT_ARN>",
        "applicableAssetTypes" : ["RedshiftViewAssetType",
 "RedshiftTableAssetType"],
        "provider": "Amazon DataZone"
}
```

Important

- The environmentIdentifier you use in the API call above should exist in the same associated account from which you are making the API call. Otherwise, the API call will not succeed.
- The IAM role ARN you use in the "authorizedPrincipals" is the role to which Amazon DataZone will grant access to after a subscribed asset is added to the subscription target. These authorized principals must belong to the same account as the environment in which the subscription target is being created.
- The value for provider field must be "Amazon DataZone" for Amazon DataZone to be able to complete subscription fulfillment.
- The database name provided in subscriptionTargetConfig should already exist in the account in which the target is being created. Amazon DataZone will not create this database. Also ensure that the manage access role has CREATE TABLE permission on this database.
- Also make sure that the roles (IAM role for the AWS Glue and the database role for Amazon Redshift) being provided as the authorized principals already exist in the environment account. For Amazon Redshift subscription targets, additional updates are required for the role being assumed while connecting to the cluster. This role must have RedshiftDbRoles tag attached to the role. The value of the tag can be a comma separated list. The value should be the database role that was provided as the authorized principal while creating the subscription target.

Step 3: Subscribe to a new table and fulfill subscription to the new target

- Once you have created the subscription target, you can subscribe to a new table and Amazon DataZone will fulfill it to the above target. For more information, see Subscribing to data.

Granting access to data

The following tasks provide details of granting access to approved subscriptions to assets in Amazon DataZone.

In Amazon DataZone, subscription requests and approved or granted subscriptions for **read** access to the assets are managed by subscription approvers. A subscription approver for an asset is determined by the publishing agreement with which this asset was published into the Amazon DataZone catalog.

Topics

Grant access to managed AWS Glue Data Catalog assets

> **Note**
>
> Access management for the AWS Glue Data Catalog assets using the AWS Lake Formation LF-TBAC method is not supported.
> Support for cross-Region sharing of assets in AWS Glue Data Catalog is not supported.

Once a subscription request to managed AWS Glue Data Catalog assets is approved, Amazon DataZone automatically adds these assets to all the existing data lake environments in the project. Amazon DataZone then grants and manages access to the approved AWS Glue Data Catalog tables on your behalf through AWS Lake Formation. For the subscriber project, assets that are granted appear in the AWS Glue Data Catalog as resources in your account. You can then use Amazon Athena to query the tables.

> **Note**
>
> If a new data lake environment is added to the project after the subscribed AWS Glue Data Catalog assets have been automatically added to the existing data lake environments, you have to manually add these subscribed AWS Glue Data Catalog assets to this new data lake environment. You can do this by choosing the **Add grant** option in the **Data** tab of the project's overview page in the Amazon DataZone data portal.

For Amazon DataZone to be able to grant access to AWS Glue Data Catalog tables, the following conditions must be met.

- The AWS Glue table must be Lake Formation-managed since Amazon DataZone grants access by managing Lake Formation permissions.
- The **Manage access role** for the data lake environment used to publish the AWS Glue Data Catalog table must have the following Lake Formation permissions:
  - `DESCRIBE` and `DESCRIBE GRANTABLE` permissions on the AWS Glue database that contains the published table.
  - `DESCRIBE`, `SELECT`, `DESCRIBE GRANTABLE`, `SELECT GRANTABLE` permissions in Lake Formation on the published table itself.

For more information, see Granting and revoking permissions on catalog resources in the *AWS Lake Formation Developer Guide*.

Grant access to managed Amazon Redshift assets

When a subscription to an Amazon Redshift table or view is approved, Amazon DataZone can automatically add the subscribed asset to all the data warehouse environments within the project, so that members of the project can query the data using the Amazon Redshift query editor link within their environments. Under the hood, Amazon DataZone, creates the necessary grants and datashares between the source and the subscription target.

The process of granting access varies depending on where the source database (publisher) and the target database (subscriber) are located.

- Same cluster, same database - if data must be shared within the same database, Amazon DataZone grants permissions directly on the source table.

- Same cluster, different database - if data must be shared across two databases within the same cluster, Amazon DataZone creates a view in the target database and permissions are granted on the created view.
- Same account different cluster - Amazon DataZone creates a datashare between the source and target cluster and creates a view on top of the shared table. Permissions are granted on the view.
- Cross-account - same as above but an additional step is required to authorize cross-account datashare on the producer cluster side and another step to associate the data share on consumer cluster side.

> **Note**
>
> If a new data warehouse environment is added to the project after the subscribed Amazon Redshift assets have been automatically added to the existing data warehouse environments, you have to manually add these subscribed Amazon Redshift assets to this new data warehouse environment. You can do this by choosing the **Add grant** option in the **Data** tab of the project's overview page in the Amazon DataZone data portal.

Make sure that your publishing and subscribing Amazon Redshift clusters meet all requirements for Amazon Redshift datashares. For more information, see Amazon Redshift Developer Guide.

> **Note**
>
> Amazon DataZone supports automatically granting subscriptions to both Amazon Redshift Cluster and Amazon Redshift Serverless assets.
> Cross-Region data sharing using Amazon Redshift is not supported.

> **Note**
>
> In the current release, Amazon DataZone can manage access to Amazon Redshift tables and views only if the source and the target Amazon Redshift clusters or workgroups are located in the AWS accounts that belong to the same AWS organization.

Grant access for approved subscriptions to unmanaged assets

Amazon DataZone enables users to publish any type of asset in the business data catalog. For some of these assets, Amazon DataZone can can automatically manage access grants. These assets are called **managed assets** and include Lake Formation-managed AWS Glue Data Catalog tables and Amazon Redshift tables and views. All other assets to which Amazon DataZone can't automatically grant subscriptions are called **unmanaged**.

Amazon DataZone provides a path for you to manage access grants for your unmanaged assets. When a subscription to an asset in the business data catalog is approved by the data owner, Amazon DataZone publishes an event in Amazon EventBridge in the your account along with all the necessary information in the payload that enables you to create the access grants between the source and the target. When you receive this event, you can trigger a custom handler which can use the information in the event to create necessary grants or permissions. Once you have granted the access, you can report back and update the status of the subscription in Amazon DataZone so that it can notify the user(s) who subscribed to the asset that they can start consuming the asset. For more information, see Working with Amazon DataZone events and notifications.

Consuming data

The following tasks provide details of consuming data that you've subscribed to in Amazon DataZone.

Topics

Query data in Amazon Athena or Amazon Redshift

In Amazon DataZone, once a subscriber has access to an asset in the catalog, they can consume it (query and analyze) using Amazon Athena or Amazon Redshift query editor v2. You must be a project owner or contributor to complete this task. Depending on the blueprints enabled in the project, Amazon DataZone provides links to Amazon Athena and/or Amazon Redshift query editor v2 on the right-hand side pane of the project page in the data portal.

1. Navigate to the Amazon DataZone data portal URL and sign in using single sign-on (SSO) or your AWS credentials. If you’re an Amazon DataZone administrator, you can navigate to the Amazon DataZone console at https://console.aws.amazon.com/datazone and sign in with the AWS account where the domain was created, then choose **Open data portal**.

2. In the Amazon DataZone data portal, choose **Browse Projects List** and then find and choose the project where you have the data that you want to analyze.
3. If the Data Lake blueprint is enabled on this project, a link to Amazon Athena is displayed in the right-hand side panel on the project's home page.

   If the Data Warehouse blueprint is enabled on this project, a link to the query editor is displayed in the right-hand side panel on the project's home page.

   > **Note**
   >
   > Blueprints are defined in the environment profile with which a project is created.

Topics

- Query data using Amazon Athena
- Query data using Amazon Redshift

Query data using Amazon Athena

Choose the Amazon Athena link to open the Amazon Athena query editor in a new tab in the browser using the project’s credentials for authentication. The Amazon DataZone project you're working with is automatically selected as the current workgroup in the query editor.

In the Amazon Athena query editor, write and run your queries. Some common tasks include:

- Query and analyze your subscribed assets
- Create new tables
- Create a table from query results (CTAS) from an external S3 bucket

Query and analyze your subscribed assets

If access to the assets that your project is subscribed to is not granted automatically by Amazon DataZone, you must be authorized to access the underlying data. For more information on how to grant access to these assets, see Grant access for approved subscriptions to unmanaged assets.

If access to the assets that your project is subscribed to is granted automatically by Amazon DataZone, you can run SQL queries on the tables and see the results in Amazon Athena. For more information about using SQL in Amazon Athena, see SQL reference for Athena.

When you navigate to the Amazon Athena query editor after choosing the Amazon Athena link in the right-hand side panel on the project's home page, a **Project** dropdown is displayed in the top-right corner of the Amazon Athena query editor and your project context is automatically selected.

You can see the following databases in the **Database** dropdown:

- A publishing database (*{environmentname}*_pub_db). The purpose of this database is to provide you with an environment where you can produce new data within the context of your project and then be able to publish this data into the Amazon DataZone catalog. Project owners and contributors have read and write access to this database. Project viewers have only read access to this database.
- A subscription database (*{environmentname}*_sub_db). The purpose of this database is to share with you the data to which you have subscribed as a project member in the Amazon DataZone catalog, and to enable you to query that data.

Create new tables

If you have connected to an external S3 bucket, you can use Amazon Athena to query and analyze the assets from an external Amazon S3 bucket. In this scenario, Amazon DataZone doesn't have permissions to grant access directly to the underlying data in the external Amazon S3 bucket, and the external Amazon S3 data created outside the project is not automatically managed in Lake Formation, and can't be managed by Amazon DataZone. An alternative is to copy the data from the external Amazon S3 bucket to a new table inside the project’s Amazon S3 bucket using a `CREATE TABLE` statement in Amazon Athena. When you run a `CREATE TABLE` query in Amazon Athena, you register your table with the AWS Glue Data Catalog.

To specify the path to your data in Amazon S3, use the `LOCATION` property, as shown in the following example:

```
CREATE EXTERNAL TABLE 'test_table'(
...
)
ROW FORMAT ...
STORED AS INPUTFORMAT ...
OUTPUTFORMAT ...
LOCATION 's3://bucketname/folder/'
```

For more information, see Table location in Amazon S3.

Create a table from query results (CTAS) from an external S3 bucket

When you subscribe to an asset, access to the underlying data is read-only. You can use Amazon Athena to create a copy of the table. In Amazon Athena, `A CREATE TABLE AS SELECT (CTAS)` query creates a new table in Amazon Athena from the results of a `SELECT` statement from another query. For information about the CTAS syntax, see CREATE TABLE AS.

The following example creates a table by copying all columns from a table:

```
CREATE TABLE new_table AS
SELECT *
FROM old_table;
```

In the following variation of the same example, your `SELECT` statement also includes a `WHERE` clause. In this case, the query selects only those rows from the table that satisfy the `WHERE` clause:

```
CREATE TABLE new_table AS
SELECT *
FROM old_table WHERE condition;
```

The following example creates a new query that runs on a set of columns from another table:

```
CREATE TABLE new_table AS
SELECT column_1, column_2, ... column_n
FROM old_table;
```

This variation of the same example creates a new table from specific columns from multiple tables:

```
CREATE TABLE new_table AS
SELECT column_1, column_2, ... column_n
FROM old_table_1, old_table_2, ... old_table_n;
```

These newly created tables are now a part of your projects' AWS Glue database, and can be made discoverable by others and shared with other Amazon DataZone projects by publishing the data as an asset to the Amazon DataZone catalog.

Query data using Amazon Redshift

In the Amazon DataZone data portal, open an environment that uses the data warehouse blueprint. Choose the **Amazon Redshift** link in the right-hand panel on the environment page. This opens a confirmation dialog with necessary details that help you establish a connection to your environmemnt's Amazon Redshift cluster or Amazon Redshift Serverless workgroup in the Amazon Redshift query editor v2.0. Once you have identified the necessary details to establish the connection, choose the **Open Amazon Redshift** button. This opens the Amazon Redshift query editor v2.0 in a new tab in the browser using temporary credentials of the Amazon DataZone environment.

In the query editor, follow the steps below depending on whether your environment is using an Amazon Redshift Serverless workgroup or an Amazon Redshift cluster.

For an Amazon Redshift Serverless workgroup

1. In the query editor, identify you Amazon DataZone environment's Amazon Redshift Serverless workgroup, right-click it and choose **Create a connection**.
2. Choose **Federated User** for authentication.
3. Provide the name of the Amazon DataZone environment's database.
4. Choose **Create connection**.

For an Amazon Redshift cluster:

1. In the query editor, identify you Amazon DataZone environment's Amazon Redshift cluster, right-click it and choose **Create a connection**.
2. Select **Temporary credentials using your IAM identity** for authentication.
3. If the above authentication method is not available, open **Account settings** by choosing the gear button in the bottom left corner, choose **Authenticate with IAM credentials** and save. This is a one-time-only setting.
4. Provide the name of the Amazon DataZone environment's database to create the connection.

5. Choose **Create connection**.

Now you can start querying against the tables and views within the Amazon Redshift cluster or Amazon Redshift Serverless workgroup configured for your Amazon DataZone environment.

Any Amazon Redshift tables or views that you have subscribed to are linked to the Amazon Redshift cluster or Amazon Redshift Serverless workgroup that is configured for the environment. You can subscribe to the tables and views as well as publish any new tables and views that you create in your environment's cluster or database.

For example, let's take a scenario in which an environment is linked to an Amazon Redshift cluster called `redshift-cluster-1` and a database called `dev` in that cluster. Using the Amazon DataZone data portal, you can query the tables and views that are added to your environment. Under the `Analytics tools` section in the right-hand side pane of the data portal, you can choose the Amazon Redshift link for this environment, which opens the query editor. You can then right-click on `redshift-cluster-1` cluster and create a connection using **Temporary credentials using your IAM identity**. Once the connection is established, you can see all the tables and views to which your environment has access under the **dev** database.

Working with Amazon DataZone events and notifications

Amazon DataZone keeps you informed of important activities within your data portal, such as subscription requests, updates, comments, and system events. Amazon DataZone provides you with this information by delivering messages in the dedicated inbox in the data portal or via the Amazon EventBridge default bus.

Topics

Working with events via the dedicated inbox in the Amazon DataZone data portal

Amazon DataZone provides a dedicated inbox in the data portal where you can see and take action on your messages. Recent messages also surface on the home page, project page, and catalog page. For example, if a user requests access to a data asset, publishing project’s owners and contributors of that asset see the request in the data portal and once an action is taken, project members of the subscribing project related to this request see the notification in the data portal. There are two types of messages:

- Tasks - these messages inform the recipient that there is action needed somewhere. They have an optional status field which you can use for tracking.
- Events - these messages are informational and have no assigned status. Events provide an audit trail of recent updates.

In Amazon DataZone, messages are generated for the following event types:

Event category	Event name	Event description	Event type
Subscription	Subscription request created	Event is generated when a subscription request is created	Task

Event category	Event name	Event description	Event type
Subscription	Subscription request accepted	Event is generated when a subscription request is accepted	Event
Subscription	Subscription request rejected	Event is generated when a subscription request is rejected	Event
Subscription	Subscription request deleted	Event is generated when a subscription request is deleted	Event
Project	Project creation succeeded	Event is generated when project creation succeeds	Event
Project membership	Project member addition succeeded	Event is generated when a new member is added to a project	Event
Project membership	Project member removal succeeded	Event is generated when a member is removed to a project	Event
Project membership	Project member role change succeeded	Event is generated a member's role in the project is changed	Event
Environment	Environment deployment started	Event is generated when a environme nt deployment is initiated	Event

Event category	Event name	Event description	Event type
Environment	Environment deployment completed	Event is generated when a environment deployment completes successfully	Event
Environment	Environment deployment failed	Event is generated when a environment deployment fails	Event
Environment	Environment deployment custom workflow initiated	Event is generated when a environment with custom workflow is initiated	Event
Data asset	Asset added to inventory	Event is generated when a new data asset is added to inventory i.e. added to catalog in draft state	Event
Data asset	Asset published	Event is generated when a new data asset is published i.e. available for subscription	Event
Data asset	Asset schema changed	Event is generated when a asset schema has changed since previous ingestion job	Event

Event category	Event name	Event description	Event type
Subscribing	Subscription created	Event is generated when someone requests to subscribe to a data asset	Task
Subscribing	Subscription approved	Event is generated when a subscription is approved by the publlishing project owner or contributor	Event
Subscribing	Subscription rejected	Event is generated when a subscription is rejected by the publlishing project owner or contributor	Event
Subscribing	Subscription deleted	Event is generated when a subscription is canceled by the subscriber	Event
Subscribing	Subscription grant requested	Event is generated when a someone requests access to an asset	Event
Subscribing	Subscription grant completed	Event is generated when a subscript ion is granted access to the asset by the publlishing project owner or contributor	Event

Event category	Event name	Event description	Event type
Subscribing	Subscription grant failed	Event is generated when a subscription grant fails	Event
Subscribing	Subscription grant revoke requested	Event is generated when a subscript ion grant revoked is initiated by the publlishing project owner or contributor	Event
Subscribing	Subscription grant revoke completed	Event is generated when a subscript ion grant revoke is completed	Event
Subscribing	Subscription grant revoke failed	Event is generated when a subscription grant revoke fails	Event
Automated business name generation	Business name generated succeeded	Eventis generated when the automated business name generated job completes successfu lly	Event
Automated business name generation	Business name generated failed	Event is generated when the automated business name generated job fails	Event
Data source run	Data source created	Event is generated when a new data source is created	Event

Event category	Event name	Event description	Event type
Data source run	Data source updated	Event is generated when an existing data source is updated	Event
Data source run	Data source run triggered	Event is generated when a data source run is initiated	Event
Data source run	Data source run succeeded	Event is generated when a data source run succeeds	Event
Data source run	Data source run failed	Event is generated when a data source run fails	Event

To view tasks in your data portal inbox, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you're an Amazon DataZone administrator, you can obtain the data portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone domain was created.
2. In the data portal, to view a pop up with the recent set of tasks, select the bell icon next to the Search bar.
3. Select View all to view all tasks. You can change views and see all events by selecting the Events tab.
4. You can filter the search by the event subject, active or inactive status, or date range.
5. Choose any individual task to navigate to the location where you can respond to the task.

To view events in your data portal inbox, complete the following steps:

1. Navigate to the Amazon DataZone data portal using the data portal URL and log in using your SSO or AWS credentials. If you're an Amazon DataZone administrator, you can obtain the data

portal URL by accessing the Amazon DataZone console at https://console.aws.amazon.com/datazone in the AWS account where the Amazon DataZone root domain was created.

2. In the data portal, to view the pop up for the recent set of events, select the bell icon next to the Search bar.
3. Select View all to view all events. You can change views and see all tasks by selecting the Tasks tab.
4. Filter the search by the event subject or date range.
5. Choose any individual event to navigate to the location where you can view details about that event.

Working with events via Amazon EventBridge default bus

In addition to sending messages to your dedicated inbox in the data portal, DataZone also sends these messages to your Amazon EventBridge default event bus in the same AWS account where your Amazon DataZone root domain is hosted. This enables event-driven automation, such as subscription fulfillment or custom integrations with other tools. You can create rules that match incoming Amazon EventBridge events and send them to Amazon EventBridge targets for processing. A single rule can send an event to multiple targets, which can then run in parallel.

Here's a sample event:

```
{
  "version": "0",
  "id": "bd3d6239-2877-f464-0572-b1d76760e085",
  "detail-type": "Subscription Request Created",
  "source": "aws.datazone",
  "account": "111111111111",
  "time": "2023-11-13T17:57:00Z",
  "region": "us-east-1",
  "resources": [],
  "detail": {
    "version": "655",
    "metadata": {
      "domain": "dzd_bc8e1ez8r2a6xz",
      "user": "44f864b8-50a1-70cc-736f-c1f763934ab7",
      "id": "5jbc0lie0sr99j",
      "version": "1",
      "typeName": "SubscriptionRequestEntityType",
      "owningProjectId": "6oy92hwk937pgn",
      "awsAccountId": "111111111111",
      "clientToken": "e781b7b5-78c5-4608-961e-3792a6c3ff0d"
    },
    "data": {
      "autoApproved": true,
      "requesterId": "44f864b8-50a1-70cc-736f-c1f763934ab7",
      "status": "PENDING",
      "subscribedListings": [
        {
          "id": "ayzstznnx4dxyf",
          "ownerProjectId": "5a3se66qm88947",
          "version": "12"
        }
      ],
      "subscribedPrincipals": [
        {
          "id": "6oy92hwk937pgn",
          "type": "PROJECT"
        }
      ]
    }
  }
}
```

The full list of detail-types supported by Amazon DataZone include:

- Subscription Request Created
- Subscription Request Accepted
- Subscription Request Rejected
- Subscription Request Deleted
- Subscription Grant Requested
- Subscription Grant Completed
- Subscription Grant Failed
- Subscription Grant Revoke Requested
- Subscription Grant Revoke Completed
- Subscription Grant Revoke Failed
- Asset Added To Inventory

- Asset Added To Catalog
- Asset Schema Changed
- Data Source Status Change
- Data Source Created
- Data Source Updated
- Data Source Run Triggered
- Data Source Run Succeeded
- Data Source Run Failed
- Domain Creation Succeeded
- Domain Creation Failed
- Domain Deletion Succeeded
- Domain Deletion Failed
- Environment Deployment Started
- Environment Deployment Completed
- Environment Deployment Failed
- Environment Deletion Started
- Environment Deletion Completed
- Environment Deletion Failed
- Project Creation Succeeded
- Project Member Addition Succeeded
- Project Member Removal Succeeded
- Project Member Role Change Succeeded
- Environment Deployment Customer Workflow Initiated
- Business Name Generation Succeeded
- Business Name Generation Failed

For more information, see Amazon EventBridge.

Security in Amazon DataZone

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from data centers and network architectures that are built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security *of* the cloud and security *in* the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to Amazon DataZone, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using Amazon DataZone. The following topics show you how to configure Amazon DataZone to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Amazon DataZone resources.

Topics

Data protection in Amazon DataZone

The AWS shared responsibility model applies to data protection in Amazon DataZone. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. You are also responsible for the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For information about data protection in Europe, see the AWS Shared Responsibility Model and GDPR blog post on the *AWS Security Blog*.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or AWS Identity and Access Management (IAM). That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.
- If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-2.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with Amazon DataZone or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

Data encryption

When granting permissions, you decide who is getting what permissions to which Amazon DataZone resources. You enable specific actions that you want to allow on those resources.

Therefore, you should grant only the permissions that are required to perform a task. Implementing least privilege access is fundamental in reducing security risk and the impact that could result from errors or malicious intent.

Encryption at rest

Amazon DataZone encrypts all your data by default with an AWS Key Management Service (AWS KMS) key that AWS owns and manages for you. You can also encrypt the data stored in the Amazon DataZone catalog using keys that you manage with AWS KMS.

When you create a domain in Amazon DataZone, you can provide encryption settings by selecting the checkbox next to **Customize encryption settings (advanced)** under **Data Encryption**, and providing a KMS key.

Encryption in transit

Amazon DataZone uses Transport Layer Security (TLS) and client-side encryption for encryption in transit. Communication with Amazon DataZone is always done over HTTPS so your data is always encrypted in transit.

Inter-network traffic privacy

To secure connections between accounts, Amazon DataZone uses service roles and IAM roles to securely connect to customer accounts and execute operations on behalf of the customer.

Topics

- Data encryption at rest for Amazon DataZone
- Using Interface VPC Endpoints for Amazon DataZone

Data encryption at rest for Amazon DataZone

Encryption of data at rest by default helps reduce the operational overhead and complexity involved in protecting sensitive data. At the same time, it enables you to build secure applications that meet strict encryption compliance and regulatory requirements.

Amazon DataZone uses default AWS-owned keys to automatically encrypt your data at rest. You can't view, manage, or audit the use of AWS owned keys. For more information, see AWS owned keys.

While you can't disable this layer of encryption or select an alternate encryption type, you can add a second layer of encryption over the existing AWS owned encryption keys by choosing a customer-managed key when you create your Amazon DataZone domains. Amazon DataZone supports the use of a symmetric customer managed keys that you can create, own, and manage to add a second layer of encryption over the existing AWS owned encryption. Because you have full control of this layer of encryption, in it you can perform the following tasks:

- Establish and maintain key policies
- Establish and maintain IAM policies and grants
- Enable and disable key policies
- Rotate key cryptographic material
- Add tags
- Create key aliases
- Schedule keys for deletion

For more information, see Customer managed keys.

> **Note**
>
> Amazon DataZone automatically enables encryption at rest using AWS owned keys to protect customer data at no charge.
> AWS KMS charges apply for using a customer managed keys. For more information about pricing, see AWS Key Management Service Pricing.

How Amazon DataZone uses grants in AWS KMS

Amazon DataZone requires three grants to use your customer managed key. When you create a Amazon DataZone domain encrypted with a customer managed key, Amazon DataZone creates grants and sub-grants on your behalf by sending CreateGrant requests to AWS KMS. Grants in AWS KMS are used to give Amazon DataZone access to a KMS key in your account. Amazon DataZone creates the following grants to use your customer managed key for the following internal operations:

One grant for encrypting your data at rest for the following operations:

- Send DescribeKey requests to AWS KMS to verify that the symmetric customer managed KMS key ID entered when creating a Amazon DataZone domain collection is valid.
- Send GenerateDataKeyrequests to AWS KMS to generate data keys encrypted by your customer managed key.
- Send Decrypt requests to AWS KMS to decrypt the encrypted data keys so that they can be used to encrypt your data.
- RetireGrant to retire the grant when domain is deleted.

Two grants for search and discovery of your data:

- Grant 2:
  - DescribeKey
  - GenerateDataKey
  - Encrypt, Decrypt, ReEncrypt
  - CreateGrant to create child grants for AWS services used internally by DataZone.
  - RetireGrant
- **Grant 3:**
  - GenerateDataKey
  - Decrypt
  - RetireGrant

You can revoke access to the grant, or remove the service's access to the customer managed key at any time. If you do, Amazon DataZone won't be able to access any of the data encrypted by the customer managed key, which affects operations that are dependent on that data. For example, if you attempt to get Data Asset details that Amazon DataZone can't access, then the operation would return an `AccessDeniedException` error.

Create a customer managed key

You can create a symmetric customer managed key by using the AWS Management Console, or the AWS KMS APIs.

To create a symmetric customer managed key, follow the steps for Creating symmetric customer managed key in the AWS Key Management Service Developer Guide.

Key policy - key policies control access to your customer managed key. Every customer managed key must have exactly one key policy, which contains statements that determine who can use the key and how they can use it. When you create your customer managed key, you can specify a key policy. For more information, see Managing access to customer managed keys in the AWS Key Management Service Developer Guide.

To use your customer managed key with your Amazon DataZone resources, the following API operations must be permitted in the key policy:

- kms:CreateGrant – adds a grant to a customer managed key. Grants control access to a specified KMS key, which allows access to grant operations Amazon DataZone requires. For more information about Using Grants, see the AWS Key Management Service Developer Guide.
- kms:DescribeKey – provides the customer managed key details to allow Amazon DataZone to validate the key.
- kms:GenerateDataKey – returns a unique symmetric data key for use outside of AWS KMS.
- kms:Decrypt – decrypts ciphertext that was encrypted by a KMS key.

The following are policy statement examples you can add for Amazon DataZone:

```
"Statement" : [
    {
      "Sid" : "Allow access to principals authorized to manage Amazon DataZone",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "arn:aws:iam::<account_id>:root"
      },
      "Action" : [
        "kms:DescribeKey",
        "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource" : "arn:aws:kms:region:<account_id>:key/key_ID",
    }
  ]
```

> **Note**
>
> Deny on the KMS policy is not applied for the resources accessed through the Amazon DataZone data portal.

For more information about specifying permissions in a policy, see the AWS Key Management Service Developer Guide.

For more information about troubleshooting key access, see the AWS Key Management Service Developer Guide.

Specifying a customer managed key for Amazon DataZone

Amazon DataZone encryption context

An encryption context is an optional set of key-value pairs that contain additional contextual information about the data.

AWS KMS uses the encryption context as additional authenticated data to support authenticated encryption. When you include an encryption context in a request to encrypt data, AWS KMS binds the encryption context to the encrypted data. To decrypt data, you include the same encryption context in the request.

Amazon DataZone uses following encryption context:

```
"encryptionContextSubset": {
    "aws:datazone:domainId": "{root-domain-uuid}"
}
```

Using encryption context for monitoring - when you use a symmetric customer managed key to encrypt Amazon DataZone, you can also use the encryption context in audit records and logs to identify how the customer managed key is being used. The encryption context also appears in logs generated by AWS CloudTrail or Amazon CloudWatch Logs.

Using encryption context to control access to your customer managed key - you can use the encryption context in key policies and IAM policies as conditions to control access to your symmetric customer managed key. You can also use encryption context constraints in a grant.

Amazon DataZone uses an encryption context constraint in grants to control access to the customer managed key in your account or region. The grant constraint requires that the operations that the grant allows use the specified encryption context.

The following are example key policy statements to grant access to a customer managed key for a specific encryption context. The condition in this policy statement requires that the grants have an encryption context constraint that specifies the encryption context.

```
{
    "Sid": "Enable DescribeKey",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
     },
     "Action": "kms:DescribeKey",
     "Resource": "*"
},{
    "Sid": "Enable Decrypt, GenerateDataKey",
     "Effect": "Allow",
     "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
     },
     "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey"
     ],
     "Resource": "*",
     "Condition": {
        "StringEquals": {
            "kms:EncryptionContext:aws:datazone:domainId": "{root-domain-uuid}"
        }
    }
}
```

Monitoring your encryption keys for Amazon DataZone

When you use an AWS KMS customer managed key with your Amazon DataZone resources, you can use AWS CloudTrail to track requests that Amazon DataZone sends to AWS KMS. The following examples are AWS CloudTrail events for `CreateGrant`, `GenerateDataKey`, `Decrypt`, and `DescribeKey` to monitor KMS operations called by Amazon DataZone to access data encrypted by your customer managed key. When you use an AWS KMS customer managed key to encrypt your Amazon DataZone domain, Amazon DataZone sends a `CreateGrant` request on your behalf to access the KMS key in your AWS account. Grants that Amazon DataZone creates are specific to the resource associated with the AWS KMS customer managed key. In addition, Amazon DataZone uses the `RetireGrant` operation to remove a grant when you delete a domain. The following example event records the `CreateGrant` operation:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAIGDTESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAIGDTESTANDEXAMPLE:Sampleuser01",
                "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-04-22T17:02:00Z"
            }
        },
        "invokedBy": "datazone.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "constraints": {
            "encryptionContextSubset": {
                "aws:datazone:domainId": "SAMPLE-root-domain-uuid"
            }
        },
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
        "operations": [
            "Decrypt",
            "GenerateDataKey",
            "RetireGrant",
            "DescribeKey"
        ],
        "granteePrincipal": "datazone.us-west-2.amazonaws.com"
    },
    "responseElements": {
        "grantId":
 "0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

Creating Data Lake environments that involve encrypted AWS Glue catalogs

In advanced use cases, when you are working with an AWS Glue catalog that is encrypted, you must grant access to the Amazon DataZone service to use your customer-managed KMS key. You can do this by updating your custom KMS policy and adding a tag to the key. To grant access to the Amazon DataZone service to work with data in an encrypted AWS Glue catalog, complete the following:

- Add the following policy to your custom KMS key. For more information, see Changing a key policy.

```
{
  "Sid": "Allow datazone environment roles to use the key",
  "Effect": "Allow",
  "Principal": {
    "AWS": "*"
  },
  "Action": [
    "kms:Decrypt",
    "kms:Describe*",
    "kms:Get*"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "aws:PrincipalArn": "arn:aws:iam::*:role/*datazone_usr*"
    }
  }
}
```

- Add the following tag to your custom KMS key. For more information, see Using tags to control access to KMS keys.

```
key: AmazonDataZoneEnvironment
value: all
```

Using Interface VPC Endpoints for Amazon DataZone

If you use Amazon Virtual Private Cloud (Amazon VPC) to host your AWS resources, you can establish a connection between your Amazon VPC and Amazon DataZone. You can use this connection with Amazon DataZone without crossing the public internet.

Amazon VPC lets you launch AWS resources in a custom virtual network. You can use a VPC to control your network settings, such as the IP address range, subnets, route tables, and network gateways. For more information about VPCs, see the Amazon VPC User Guide.

To connect your Amazon VPC to Amazon DataZone, you must first define an interface VPC endpoint, which lets you connect your VPC to other AWS services. The endpoint provides reliable, scalable connectivity, without requiring an internet gateway, network address translation (NAT) instance, or VPN connection. For more information and detailed steps on how to create a VPC endpoint, see Interface VPC Endpoints (AWS PrivateLink) in the Amazon VPC User Guide.

⚠ **Important**

In VPC, an endpoint policy is a resource-based policy that you can attach to a VPC endpoint to control which AWS principals can use the endpoint to access an AWS service.
In the current release of Amazon DataZone, the use of endpoint policies is not supported for establishing and using connections between your Amazon VPC and Amazon DataZone. Amazon DataZone access management relies on RAM configuration and IAM principal policies that are defined at the service level.

Authorization in Amazon DataZone

Amazon DataZone’s interface consists of a management console within AWS and an off-console web application (data portal).

The Amazon DataZone management console can be used by AWS administrators for top-level-resource APIs, including creating and managing domains, AWS account associations for these domains, and data sources for which you want to delegate access management to Amazon DataZone. You can use the Amazon DataZone management console to manage all of the IAM roles and configuration needed to delegate access management control to the Amazon DataZone service for their explicitly configured AWS accounts. The Amazon DataZone data portal is a first-party AWS Identity Center application for SSO users. If enabled, the console can also be used by authorized IAM principals to federate into the data portal instead of using an SSO identity.

Amazon DataZone's data portal is designed to be used principally by AWS IAM Identity Center-authenticated users to manage access to data and perform data publishing, discovery, subscription, and analytics tasks.

Authorization in the Amazon DataZone console

The Amazon DataZone console authorization model uses IAM authorization. The console is used by administrators primarily for setup. Amazon DataZone uses the concept of a domain administrator

AWS account, and member AWS accounts, and the console is used from all of these accounts to build the trust relationships while respecting AWS Organization boundaries.

Authorization in the Amazon DataZone portal

The Amazon DataZone data portal authorization model is a hierarchical ACL with static role archetypes (profiles) that include administrators and viewers. For example, users can have a profile of administrator or user. At the level of a domain, they may have a domain user designation of data owner. At the level of a project, a user can be an owner or contributor. These profiles can be configured as one of two types: users and groups. These profiles are then associated with domains and projects, and the state for these permissions is stored in an association table.

Within this authorization model, Amazon DataZone allows users to manage user and group permissions. Users manage project membership, request membership to projects, and approve memberships. Users publish data, define data subscription approvers, subscribe to data, and approve subscriptions.

Users perform data analytics in specific projects when their data portal client requests IAM session credentials that Amazon DataZone generates based on the user's effective profile in the specific project context. This session is scoped both to the user's permissions and also the specific project's resources. Users then drop into Athena or Redshift to query the relevant data, and all of the underlying IAM work is completely abstracted away.

Amazon DataZone profiles and roles

Once a user is authenticated, the authenticated context maps to a user profile ID. This user profile can have multiple, different associations (project owner, domain administrator, etc.) which is used for authorizing users. Each association (for example, project owner, domain administrator, etc.) has permissions for certain activities based on the context. For example, a user that has a domain admin association can create additional domains, can assign other domain administrators to the domain, and can create project templates within their domain. A project owner can add or remove project members for their project, they can create publishing agreements with a domain, and publish assets to a domain.

Controlling access to Amazon DataZone resources using IAM

You need AWS Identity and Access Management (IAM) to complete the following security-related tasks:

- Create users and groups under your AWS account.
- Assign unique security credentials to each user under your AWS account.
- Control each user's permissions to perform tasks with AWS resources.
- Allow the users in another AWS account to share your AWS resources.
- Create roles for your AWS account and define the users or services that can assume them.
- Use existing identities for your enterprise to grant permissions to perform tasks using AWS resources

For more information about IAM, see the following:

- AWS Identity and Access Management (IAM)
- Getting started
- IAM User Guide

The following sections describe the policies and permissions that are required to set up Amazon DataZone and its components, such as domains (including the domain), associated accounts, projects, and data sources. For more information, see Amazon DataZone terminology and concepts.

Contents

AWS managed policies for Amazon DataZone

An AWS managed policy is a standalone policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases so that you can start assigning permissions to users, groups, and roles.

Keep in mind that AWS managed policies might not grant least-privilege permissions for your specific use cases because they're available for all AWS customers to use. We recommend that you reduce permissions further by defining customer managed policies that are specific to your use cases.

You cannot change the permissions defined in AWS managed policies. If AWS updates the permissions defined in an AWS managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. AWS is most likely to update an AWS managed policy when a new AWS service is launched or new API operations become available for existing services.

For more information, see AWS managed policies in the *IAM User Guide*.

Contents

AWS managed policy: AmazonDataZoneFullAccess

You can attach the `AmazonDataZoneFullAccess` policy to your IAM identities.

This policy provides full access to Amazon DataZone via the AWS Management Console.

Permissions details

This policy includes the following permissions:

- `datazone` – grants principals full access to Amazon DataZone via the AWS Management Console.

- `kms` – Allows principals to list aliases and describe keys.
- `s3` – Allows principals to choose existing or create new S3 buckets to store Amazon DataZone data.
- `ram` – Allows principals to share Amazon DataZone domains across AWS accounts.
- `iam` – Allows principals to list and pass roles and get policies.
- `sso` – Allows principals to obtain the regions where AWS IAM Identity Center is enabled.
- `secretsmanager` – Allows principals to create, tag, and list secrets with a specific prefix.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonDataZoneStatement",
            "Effect": "Allow",
            "Action": [
                "datazone:*"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "ReadOnlyStatement",
            "Effect": "Allow",
            "Action": [
                "kms:DescribeKey",
                "kms:ListAliases",
                "iam:ListRoles",
                "sso:DescribeRegisteredRegions",
                "s3:ListAllMyBuckets",
                "redshift:DescribeClusters",
                "redshift-serverless:ListWorkgroups",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "secretsmanager:ListSecrets"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "BucketReadOnlyStatement",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": "arn:aws:s3:::*"
        },
        {
            "Sid": "CreateBucketStatement",
            "Effect": "Allow",
            "Action": "s3:CreateBucket",
            "Resource": "arn:aws:s3:::amazon-datazone*"
        },
        {
            "Sid": "RamCreateResourceStatement",
            "Effect": "Allow",
            "Action": [
                "ram:CreateResourceShare"
            ],
            "Resource": "*",
            "Condition": {
                "StringEqualsIfExists": {
                    "ram:RequestedResourceType": "datazone:Domain"
                }
            }
        },
        {
            "Sid": "RamResourceStatement",
            "Effect": "Allow",
            "Action": [
                "ram:DeleteResourceShare",
                "ram:AssociateResourceShare",
                "ram:DisassociateResourceShare",
                "ram:RejectResourceShareInvitation"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ram:ResourceShareName": [
                        "DataZone*"
                    ]
                }
            }
        },
        {
            "Sid": "RamResourceReadOnlyStatement",
            "Effect": "Allow",
            "Action": [
                "ram:GetResourceShares",
                "ram:GetResourceShareInvitations",
                "ram:GetResourceShareAssociations",
                "ram:ListResourceSharePermissions"
            ],
            "Resource": "*"
        },
        {
            "Sid": "IAMPassRoleStatement",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/AmazonDataZone*",
                "arn:aws:iam::*:role/service-role/AmazonDataZone*"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:passedToService": "datazone.amazonaws.com"
                }
            }
        },
        {
            "Sid": "IAMGetPolicyStatement",
            "Effect": "Allow",
            "Action": "iam:GetPolicy",
            "Resource": [
                "arn:aws:iam::*:policy/service-role/
AmazonDataZoneRedshiftAccessPolicy*"
            ]
        },
        {
            "Sid": "DataZoneTagOnCreateDomainProjectTags",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:TagResource"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:AmazonDataZone-*",
            "Condition": {
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": [
                        "AmazonDataZoneDomain",
                        "AmazonDataZoneProject"
                    ]
                },
                "StringLike": {
                    "aws:RequestTag/AmazonDataZoneDomain": "dzd_*",
                    "aws:ResourceTag/AmazonDataZoneDomain": "dzd_*"
                }
            }
        },
        {
            "Sid": "DataZoneTagOnCreate",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:TagResource"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:AmazonDataZone-*",
            "Condition": {
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": [
                        "AmazonDataZoneDomain"
                    ]
                },
                "StringLike": {
                    "aws:RequestTag/AmazonDataZoneDomain": "dzd_*",
                    "aws:ResourceTag/AmazonDataZoneDomain": "dzd_*"
                }
            }
        },
        {
            "Sid": "CreateSecretStatement",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:CreateSecret"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:AmazonDataZone-*",
            "Condition": {
                "StringLike": {
                    "aws:RequestTag/AmazonDataZoneDomain": "dzd_*"
                }
            }
        }
    ]
}
```

Policy considerations and limitations

There are certain functionalities that the `AmazonDataZoneFullAccess` policy doesn't cover.

- If you create an Amazon DataZone domain with your own AWS KMS key, you must have the permissions to `kms:CreateGrant` for domain creation to succeed, and to `kms:GenerateDataKey`, `kms:Decrypt` for that key to invoke other Amazon DataZone APIs such as `listDataSources` and `createDataSource`. And you must also have the permissions to `kms:CreateGrant`, `kms:Decrypt`, `kms:GenerateDataKey`, and `kms:DescribeKey` in the resource policy of that key.

  If you use the default service-owned KMS key, then this isn't required.

  For more information, see AWS Key Management Service.
- If you want to use *create* and *update* role functionalities within the Amazon DataZone console, you must have administrator privileges or have the required IAM permissions to create IAM roles and create/update policies. The required permissions include `iam:CreateRole`, `iam:CreatePolicy`, `iam:CreatePolicyVersion`, `iam:DeletePolicyVersion`, and `iam:AttachRolePolicy` permissions.
- If you create a new domain in Amazon DataZone with AWS IAM Identity Center users login activated, or if you activate it for an existing domain in Amazon DataZone, you must have permissions to the following: `sso:CreateManagedApplicationInstance`, `sso:DeleteManagedApplicationInstance`, and `sso:PutApplicationAssignmentConfiguration`.
- In order to accept an AWS account association request in Amazon DataZone, you must have the `ram:AcceptResourceShareInvitation` permission.

AWS managed policy: AmazonDataZoneFullUserAccess

This policy grants full access to Amazon DataZone, but it doesn't allow the management of domains, users, or associated accounts.

Permissions details

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AmazonDataZoneUserOperations",
            "Effect": "Allow",
            "Action": [
                "datazone:PostTimeSeriesDataPoints",
                "datazone:ListTimeSeriesDataPoints",
                "datazone:GetTimeSeriesDataPoint",
                "datazone:DeleteTimeSeriesDataPoints",
                "datazone:GetDomain",
                "datazone:CreateFormType",
                "datazone:GetFormType",
                "datazone:GetIamPortalLoginUrl",
                "datazone:SearchUserProfiles",
                "datazone:SearchGroupProfiles",
                "datazone:GetUserProfile",
                "datazone:GetGroupProfile",
                "datazone:ListGroupsForUser",
                "datazone:DeleteFormType",
                "datazone:CreateAssetType",
                "datazone:GetAssetType",
                "datazone:DeleteAssetType",
                "datazone:CreateGlossary",
                "datazone:GetGlossary",
                "datazone:DeleteGlossary",
                "datazone:UpdateGlossary",
                "datazone:CreateGlossaryTerm",
                "datazone:GetGlossaryTerm",
                "datazone:DeleteGlossaryTerm",
                "datazone:UpdateGlossaryTerm",
                "datazone:CreateAsset",
                "datazone:GetAsset",
                "datazone:DeleteAsset",
                "datazone:CreateAssetRevision",
                "datazone:ListAssetRevisions",
                "datazone:AcceptPredictions",
                "datazone:RejectPredictions",
                "datazone:Search",
                "datazone:SearchTypes",
                "datazone:CreateListingChangeSet",
                "datazone:DeleteListing",
                "datazone:SearchListings",
                "datazone:GetListing",
                "datazone:CreateDataSource",
                "datazone:GetDataSource",
                "datazone:DeleteDataSource",
                "datazone:UpdateDataSource",
                "datazone:ListDataSources",
                "datazone:StartDataSourceRun",
                "datazone:GetDataSourceRun",
                "datazone:ListDataSourceRuns",
                "datazone:ListDataSourceRunActivities",
                "datazone:ListEnvironmentBlueprintConfigurations",
                "datazone:CreateEnvironmentBlueprint",
                "datazone:GetEnvironmentBlueprint",
                "datazone:DeleteEnvironmentBlueprint",
                "datazone:UpdateEnvironmentBlueprint",
                "datazone:ListEnvironmentBlueprints",
                "datazone:CreateProject",
                "datazone:UpdateProject",
                "datazone:GetProject",
                "datazone:DeleteProject",
                "datazone:ListProjects",
                "datazone:CreateProjectMembership",
                "datazone:DeleteProjectMembership",
                "datazone:ListProjectMemberships",
                "datazone:CreateEnvironmentProfile",
                "datazone:GetEnvironmentProfile",
                "datazone:UpdateEnvironmentProfile",
                "datazone:DeleteEnvironmentProfile",
                "datazone:ListEnvironmentProfiles",
                "datazone:CreateEnvironment",
                "datazone:GetEnvironment",
                "datazone:DeleteEnvironment",
                "datazone:UpdateEnvironment",
                "datazone:UpdateEnvironmentDeploymentStatus",
                "datazone:ListEnvironments",
                "datazone:ListAccountEnvironments",
                "datazone:GetEnvironmentActionLink",
                "datazone:GetEnvironmentCredentials",
                "datazone:GetSubscriptionTarget",
                "datazone:DeleteSubscriptionTarget",
                "datazone:ListSubscriptionTargets",
                "datazone:CreateSubscriptionRequest",
                "datazone:AcceptSubscriptionRequest",
                "datazone:UpdateSubscriptionRequest",
                "datazone:ListWarehouseMetadata",
                "datazone:RejectSubscriptionRequest",
                "datazone:GetSubscriptionRequestDetails",
                "datazone:ListSubscriptionRequests",
                "datazone:DeleteSubscriptionRequest",
                "datazone:GetSubscription",
                "datazone:CancelSubscription",
                "datazone:GetSubscriptionEligibility",
                "datazone:ListSubscriptions",
                "datazone:RevokeSubscription",
                "datazone:CreateSubscriptionGrant",
                "datazone:DeleteSubscriptionGrant",
                "datazone:GetSubscriptionGrant",
                "datazone:ListSubscriptionGrants",
                "datazone:UpdateSubscriptionGrantStatus",
                "datazone:ListNotifications",
                "datazone:StartMetadataGenerationRun",
                "datazone:GetMetadataGenerationRun",
                "datazone:CancelMetadataGenerationRun",
                "datazone:ListMetadataGenerationRuns",
                "datazone:ListLineageNodeHistory",
                "datazone:GetLineageNode",
                "datazone:CreateAssetFilter",
                "datazone:DeleteAssetFilter",
                "datazone:GetAssetFilter",
                "datazone:ListAssetFilters",
                "datazone:UpdateAssetFilter"
            ],
            "Resource": "*"
        },
        {
            "Sid": "RAMResourceShareOperations",
            "Effect": "Allow",
            "Action": "ram:GetResourceShareAssociations",
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AmazonDataZoneCustomEnvironmentDeploymentPolicy

You can use this policy to update the configuration of environments that are created using custom blueprints. This policy can also be used to create Amazon DataZone subscription targets and data sources.

Permissions details

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AmazonDataZoneCustomEnvironment",
   "Effect": "Allow",
   "Action": [
    "datazone:ListAssociatedAccounts",
    "datazone:GetAccountAssociation",
    "datazone:GetEnvironment",
    "datazone:GetEnvironmentProfile",
    "datazone:GetEnvironmentBlueprint",
    "datazone:GetProject",
    "datazone:UpdateEnvironmentConfiguration",
    "datazone:UpdateEnvironmentDeploymentStatus",
    "datazone:CreateSubscriptionTarget",
    "datazone:CreateDataSource"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS managed policy: AmazonDataZoneEnvironmentRolePermissionsBoundary

Note

This policy is a *permissions boundary*. A permissions boundary sets the maximum permissions that an identity-based policy can grant to an IAM entity. You should not use and attach Amazon DataZone permissions boundary policies on your own. Amazon DataZone permissions boundary policies should only be attached to Amazon DataZone managed roles. For more information on permissions boundaries, see Permissions boundaries for IAM entities in the IAM User Guide.

When you create an environment via the Amazon DataZone data portal, Amazon DataZone applies this permissions boundary to the IAM roles that are produced during environment creation. The permissions boundary limits the scope of the roles that Amazon DataZone creates and any roles that you add.

Amazon DataZone uses the `AmazonDataZoneEnvironmentRolePermissionsBoundary` managed policy to limit the provisioned IAM principal to which it is attached. The principals might take the form of the user roles that Amazon DataZone can assume on behalf of interactive enterprise users or analytic services (AWS Glue, for example), and then conduct actions to process data such as reading and writing from Amazon S3 or running AWS Glue crawler.

The `AmazonDataZoneEnvironmentRolePermissionsBoundary` policy grants read and write access for Amazon DataZone to services such as AWS Glue, Amazon S3, AWS Lake Formation, Amazon Redshift, and Amazon Athena. The policy also gives read and write permissions to some infrastructure resources that are required to use these services such as network interfaces and AWS KMS keys.

Amazon DataZone applies the `AmazonDataZoneEnvironmentRolePermissionsBoundary` AWS managed policy as a permissions boundary for all Amazon DataZone environment roles (owner and contributor). This permissions boundary restricts these roles to only allow access to the required resources and actions necessary for an environment.

The boundary includes the following JSON statements:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateGlueConnection",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateTags",
        "ec2:DeleteTags"
      ],
      "Resource": [
        "arn:aws:ec2:*:*:network-interface/*"
      ],
      "Condition": {
        "ForAllValues:StringEquals": {
          "aws:TagKeys": [
            "aws-glue-service-resource"
          ]
        }
      }
    },
    {
      "Sid": "GlueOperations",
      "Effect": "Allow",
      "Action": [
        "glue:*DataQuality*",
        "glue:BatchCreatePartition",
        "glue:BatchDeleteConnection",
        "glue:BatchDeletePartition",
        "glue:BatchDeleteTable",
        "glue:BatchDeleteTableVersion",
        "glue:BatchGetJobs",
        "glue:BatchGetWorkflows",
        "glue:BatchStopJobRun",
        "glue:BatchUpdatePartition",
        "glue:CreateBlueprint",
        "glue:CreateConnection",
        "glue:CreateCrawler",
        "glue:CreateDatabase",
        "glue:CreateJob",
        "glue:CreatePartition",
        "glue:CreatePartitionIndex",
        "glue:CreateTable",
        "glue:CreateWorkflow",
        "glue:DeleteBlueprint",
        "glue:DeleteColumnStatisticsForPartition",
        "glue:DeleteColumnStatisticsForTable",
        "glue:DeleteConnection",
        "glue:DeleteCrawler",
        "glue:DeleteJob",
        "glue:DeletePartition",
        "glue:DeletePartitionIndex",
        "glue:DeleteTable",
        "glue:DeleteTableVersion",
        "glue:DeleteWorkflow",
        "glue:GetColumnStatisticsForPartition",
        "glue:GetColumnStatisticsForTable",
        "glue:GetConnection",
        "glue:GetDatabase",
        "glue:GetDatabases",
        "glue:GetTable",
        "glue:GetTables",
        "glue:GetPartition",
        "glue:GetPartitions",
        "glue:ListSchemas",
        "glue:ListJobs",
        "glue:NotifyEvent",
        "glue:PutWorkflowRunProperties",
        "glue:ResetJobBookmark",
        "glue:ResumeWorkflowRun",
        "glue:SearchTables",
        "glue:StartBlueprintRun",
        "glue:StartCrawler",
        "glue:StartCrawlerSchedule",
        "glue:StartJobRun",
        "glue:StartWorkflowRun",
        "glue:StopCrawler",
        "glue:StopCrawlerSchedule",
        "glue:StopWorkflowRun",
        "glue:UpdateBlueprint",
        "glue:UpdateColumnStatisticsForPartition",
        "glue:UpdateColumnStatisticsForTable",
        "glue:UpdateConnection",
        "glue:UpdateCrawler",
        "glue:UpdateCrawlerSchedule",
        "glue:UpdateDatabase",
        "glue:UpdateJob",
        "glue:UpdatePartition",
        "glue:UpdateTable",
        "glue:UpdateWorkflow"
      ],
      "Resource": "*",
      "Condition": {
        "Null": {
          "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
        }
      }
    },
    {
      "Sid": "PassRole",
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/datazone*"
      ],
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "glue.amazonaws.com"
        }
      }
    },
    {
      "Sid": "SameAccountKmsOperations",
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:Decrypt",
        "kms:ListKeys"
      ],
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "aws:ResourceAccount": "${aws:PrincipalAccount}"
        }
      }
    },
    {
      "Sid": "KmsOperationsWithResourceTag",
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:Decrypt",
        "kms:ListKeys",
        "kms:Encrypt",
        "kms:GenerateDataKey",
        "kms:Verify",
        "kms:Sign"
      ],
      "Resource": "*",
      "Condition": {
        "Null": {
          "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
        }
      }
    },
    {
      "Sid": "AnalyticsOperations",
      "Effect": "Allow",
      "Action": [
        "datazone:*",
        "sqlworkbench:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "QueryOperations",
      "Effect": "Allow",
      "Action": [
        "athena:BatchGetNamedQuery",
        "athena:BatchGetPreparedStatement",
        "athena:BatchGetQueryExecution",
        "athena:CreateNamedQuery",
        "athena:CreateNotebook",
        "athena:CreatePreparedStatement",
        "athena:CreatePresignedNotebookUrl",
        "athena:DeleteNamedQuery",
        "athena:DeleteNotebook",
        "athena:DeletePreparedStatement",
        "athena:ExportNotebook",
        "athena:GetDatabase",
        "athena:GetDataCatalog",
        "athena:GetNamedQuery",
        "athena:GetPreparedStatement",
        "athena:GetQueryExecution",
        "athena:GetQueryResults",
        "athena:GetQueryRuntimeStatistics",
        "athena:GetTableMetadata",
        "athena:GetWorkGroup",
        "athena:ImportNotebook",
        "athena:ListDatabases",
        "athena:ListDataCatalogs",
        "athena:ListEngineVersions",
        "athena:ListNamedQueries",
        "athena:ListPreparedStatements",
        "athena:ListQueryExecutions",
        "athena:ListTableMetadata",
        "athena:ListTagsForResource",
        "athena:ListWorkGroups",
        "athena:StartCalculationExecution",
        "athena:StartQueryExecution",
        "athena:StartSession",
        "athena:StopCalculationExecution",
        "athena:StopQueryExecution",
        "athena:TerminateSession",
        "athena:UpdateNamedQuery",
        "athena:UpdateNotebook",
        "athena:UpdateNotebookMetadata",
        "athena:UpdatePreparedStatement",
        "ec2:CreateNetworkInterface",
        "ec2:DeleteNetworkInterface",
        "ec2:Describe*",
        "glue:BatchCreatePartition",
        "glue:BatchDeletePartition",
        "glue:BatchDeleteTable",
        "glue:BatchDeleteTableVersion",
        "glue:BatchGetJobs",
        "glue:BatchGetPartition",
        "glue:BatchGetWorkflows",
        "glue:BatchUpdatePartition",
        "glue:CreateBlueprint",
        "glue:CreateConnection",
        "glue:CreateCrawler",
        "glue:CreateDatabase",
        "glue:CreateJob",
        "glue:CreatePartition",
        "glue:CreatePartitionIndex",
        "glue:CreateTable",
        "glue:CreateWorkflow",
        "glue:DeleteColumnStatisticsForPartition",
        "glue:DeleteColumnStatisticsForTable",
        "glue:DeletePartition",
        "glue:DeletePartitionIndex",
        "glue:DeleteTable",
        "glue:DeleteTableVersion",
        "glue:GetColumnStatisticsForPartition",
        "glue:GetColumnStatisticsForTable",
        "glue:GetConnection",
        "glue:GetDatabase",
        "glue:GetDatabases",
        "glue:GetTable",
        "glue:GetTables",
        "glue:GetPartition",
        "glue:GetPartitions",
        "glue:ListSchemas",
        "glue:ListJobs",
        "glue:NotifyEvent",
        "glue:SearchTables",
        "glue:UpdateColumnStatisticsForPartition",
        "glue:UpdateColumnStatisticsForTable",
        "glue:UpdateDatabase",
        "glue:UpdatePartition",
        "glue:UpdateTable",
        "iam:GetRole",
        "iam:GetRolePolicy",
        "iam:ListGroups",
        "iam:ListRolePolicies",
        "iam:ListRoles",
        "iam:ListUsers",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams",
        "logs:DescribeMetricFilters",
        "logs:DescribeQueries",
        "logs:DescribeQueryDefinitions",
        "logs:DescribeMetricFilters",
        "logs:StartQuery",
        "logs:StopQuery",
        "logs:GetLogEvents",
        "logs:GetLogGroupFields",
        "logs:GetQueryResults",
        "logs:GetLogRecord",
        "logs:PutLogEvents",
        "logs:CreateLogStream",
        "logs:FilterLogEvents",
        "lakeformation:GetDataAccess",
        "lakeformation:GetDataLakeSettings",
        "lakeformation:GetResourceLFTags",
        "lakeformation:ListPermissions",
        "redshift-data:ListTables",
        "redshift-data:DescribeTable",
        "redshift-data:ListSchemas",
        "redshift-data:ListDatabases",
        "redshift-data:ExecuteStatement",
        "redshift-data:GetStatementResult",
        "redshift-data:DescribeStatement",
        "redshift:CreateClusterUser",
        "redshift:DescribeClusters",
        "redshift:DescribeDataShares",
        "redshift:GetClusterCredentials",
        "redshift:GetClusterCredentialsWithIAM",
        "redshift:JoinGroup",
        "redshift-serverless:ListNamespaces",
        "redshift-serverless:ListWorkgroups",
        "redshift-serverless:GetNamespace",
        "redshift-serverless:GetWorkgroup",
        "redshift-serverless:GetCredentials",
        "secretsmanager:ListSecrets",
        "tag:GetResources"
      ],
      "Resource": "*"
    },
    {
      "Sid": "QueryOperationsWithResourceTag",
      "Effect": "Allow",
      "Action": [
        "athena:GetQueryResultsStream"
      ],
      "Resource": "*",
      "Condition": {
        "Null": {
          "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
        }
      }
    },
    {
      "Sid": "SecretsManagerOperationsWithTagKeys",
      "Effect": "Allow",
      "Action": [
        "secretsmanager:CreateSecret",
        "secretsmanager:TagResource"
      ],
      "Resource": "arn:aws:secretsmanager:*:*:secret:AmazonDataZone-*",
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/AmazonDataZoneDomain": "*",
          "aws:ResourceTag/AmazonDataZoneProject": "*"
        },
        "Null": {
          "aws:TagKeys": "false"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": [
            "AmazonDataZoneDomain",
            "AmazonDataZoneProject"
          ]
        }
      }
    },
    {
      "Sid": "DataZoneS3Buckets",
      "Effect": "Allow",
      "Action": [
        "s3:AbortMultipartUpload",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion",
        "s3:GetObject",
        "s3:PutObject",
        "s3:PutObjectRetention",
        "s3:ReplicateObject",
        "s3:RestoreObject"
      ],
      "Resource": [
        "arn:aws:s3:::*/datazone/*"
      ]
    },
    {
      "Sid": "DataZoneS3BucketLocation",
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation"
      ],
      "Resource": "*"
    },
    {
      "Sid": "ListDataZoneS3Bucket",
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
          "StringLike": {
            "s3:prefix": [
              "*/datazone/*",
              "datazone/*"
            ]
          }
        }
    },
    {
      "Sid": "NotDeniedOperations",
      "Effect": "Deny",
      "NotAction": [
        "datazone:*",
        "sqlworkbench:*",
        "athena:BatchGetNamedQuery",
        "athena:BatchGetPreparedStatement",
        "athena:BatchGetQueryExecution",
        "athena:CreateNamedQuery",
        "athena:CreateNotebook",
        "athena:CreatePreparedStatement",
        "athena:CreatePresignedNotebookUrl",
        "athena:DeleteNamedQuery",
        "athena:DeleteNotebook",
        "athena:DeletePreparedStatement",
        "athena:ExportNotebook",
        "athena:GetDatabase",
        "athena:GetDataCatalog",
        "athena:GetNamedQuery",
        "athena:GetPreparedStatement",
        "athena:GetQueryExecution",
        "athena:GetQueryResults",
        "athena:GetQueryResultsStream",
        "athena:GetQueryRuntimeStatistics",
        "athena:GetTableMetadata",
        "athena:GetWorkGroup",
        "athena:ImportNotebook",
        "athena:ListDatabases",
        "athena:ListDataCatalogs",
        "athena:ListEngineVersions",
        "athena:ListNamedQueries",
        "athena:ListPreparedStatements",
        "athena:ListQueryExecutions",
        "athena:ListTableMetadata",
        "athena:ListTagsForResource",
        "athena:ListWorkGroups",
        "athena:StartCalculationExecution",
        "athena:StartQueryExecution",
        "athena:StartSession",
        "athena:StopCalculationExecution",
        "athena:StopQueryExecution",
        "athena:TerminateSession",
        "athena:UpdateNamedQuery",
        "athena:UpdateNotebook",
        "athena:UpdateNotebookMetadata",
        "athena:UpdatePreparedStatement",
        "ec2:CreateNetworkInterface",
        "ec2:CreateTags",
        "ec2:DeleteNetworkInterface",
        "ec2:DeleteTags",
        "ec2:Describe*",
        "glue:*DataQuality*",
        "glue:BatchCreatePartition",
        "glue:BatchDeleteConnection",
        "glue:BatchDeletePartition",
        "glue:BatchDeleteTable",
        "glue:BatchDeleteTableVersion",
        "glue:BatchGetJobs",
        "glue:BatchGetPartition",
        "glue:BatchGetWorkflows",
        "glue:BatchStopJobRun",
        "glue:BatchUpdatePartition",
        "glue:CreateBlueprint",
        "glue:CreateConnection",
        "glue:CreateCrawler",
        "glue:CreateDatabase",
        "glue:CreateJob",
        "glue:CreatePartition",
        "glue:CreatePartitionIndex",
        "glue:CreateTable",
        "glue:CreateWorkflow",
        "glue:DeleteBlueprint",
        "glue:DeleteColumnStatisticsForPartition",
        "glue:DeleteColumnStatisticsForTable",
        "glue:DeleteConnection",
        "glue:DeleteCrawler",
        "glue:DeleteJob",
        "glue:DeletePartition",
        "glue:DeletePartitionIndex",
        "glue:DeleteTable",
        "glue:DeleteTableVersion",
        "glue:DeleteWorkflow",
        "glue:GetColumnStatisticsForPartition",
        "glue:GetColumnStatisticsForTable",
        "glue:GetConnection",
        "glue:GetDatabase",
        "glue:GetDatabases",
        "glue:GetTable",
        "glue:GetTables",
        "glue:GetPartition",
        "glue:GetPartitions",
        "glue:ListSchemas",
        "glue:ListJobs",
        "glue:NotifyEvent",
        "glue:PutWorkflowRunProperties",
        "glue:ResetJobBookmark",
        "glue:ResumeWorkflowRun",
        "glue:SearchTables",
        "glue:StartBlueprintRun",
        "glue:StartCrawler",
        "glue:StartCrawlerSchedule",
        "glue:StartJobRun",
        "glue:StartWorkflowRun",
        "glue:StopCrawler",
        "glue:StopCrawlerSchedule",
        "glue:StopWorkflowRun",
        "glue:UpdateBlueprint",
        "glue:UpdateColumnStatisticsForPartition",
        "glue:UpdateColumnStatisticsForTable",
        "glue:UpdateConnection",
        "glue:UpdateCrawler",
        "glue:UpdateCrawlerSchedule",
        "glue:UpdateDatabase",
        "glue:UpdateJob",
        "glue:UpdatePartition",
        "glue:UpdateTable",
        "glue:UpdateWorkflow",
        "iam:GetRole",
        "iam:GetRolePolicy",
        "iam:List*",
        "iam:PassRole",
        "kms:DescribeKey",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:GenerateDataKey",
        "kms:ListKeys",
        "kms:Verify",
        "kms:Sign",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams",
        "logs:DescribeMetricFilters",
        "logs:DescribeQueries",
        "logs:DescribeQueryDefinitions",
        "logs:StartQuery",
        "logs:StopQuery",
        "logs:GetLogEvents",
        "logs:GetLogGroupFields",
        "logs:GetQueryResults",
        "logs:GetLogRecord",
        "logs:PutLogEvents",
        "logs:CreateLogStream",
        "logs:FilterLogEvents",
        "lakeformation:GetDataAccess",
        "lakeformation:GetDataLakeSettings",
        "lakeformation:GetResourceLFTags",
        "lakeformation:ListPermissions",
        "redshift-data:ListTables",
        "redshift-data:DescribeTable",
        "redshift-data:ListSchemas",
        "redshift-data:ListDatabases",
        "redshift-data:ExecuteStatement",
        "redshift-data:GetStatementResult",
        "redshift-data:DescribeStatement",
        "redshift:CreateClusterUser",
        "redshift:DescribeClusters",
        "redshift:DescribeDataShares",
        "redshift:GetClusterCredentials",
        "redshift:GetClusterCredentialsWithIAM",
        "redshift:JoinGroup",
        "redshift-serverless:ListNamespaces",
        "redshift-serverless:ListWorkgroups",
        "redshift-serverless:GetNamespace",
        "redshift-serverless:GetWorkgroup",
        "redshift-serverless:GetCredentials",
        "s3:AbortMultipartUpload",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion",
        "s3:GetObject",
        "s3:GetBucketLocation",
        "s3:ListBucket",
        "s3:PutObject",
        "s3:PutObjectRetention",
        "s3:ReplicateObject",
        "s3:RestoreObject",
        "secretsmanager:CreateSecret",
        "secretsmanager:ListSecrets",
        "secretsmanager:TagResource",
        "tag:GetResources"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

AWS managed policy: AmazonDataZoneRedshiftGlueProvisioningPolicy

The AmazonDataZoneRedshiftGlueProvisioningPolicy policy grants Amazon DataZone the permissions required to interoperate with AWS Glue and Amazon Redshift.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AmazonDataZonePermissionsToCreateEnvironmentRole",
   "Effect": "Allow",
   "Action": [
    "iam:CreateRole",
    "iam:DetachRolePolicy",
    "iam:DeleteRolePolicy",
    "iam:AttachRolePolicy",
    "iam:PutRolePolicy"
   ],
   "Resource": "arn:aws:iam::*:role/datazone*",
   "Condition": {
    "StringEquals": {
     "iam:PermissionsBoundary": "arn:aws:iam::aws:policy/
AmazonDataZoneEnvironmentRolePermissionsBoundary",
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "IamPassRolePermissions",
   "Effect": "Allow",
   "Action": [
    "iam:PassRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/datazone*"
   ],
   "Condition": {
    "StringEquals": {
     "iam:PassedToService": [
      "glue.amazonaws.com",
      "lakeformation.amazonaws.com"
     ],
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AmazonDataZonePermissionsToManageCreatedEnvironmentRole",
   "Effect": "Allow",
   "Action": [
    "iam:DeleteRole",
    "iam:GetRole"
   ],
   "Resource": "arn:aws:iam::*:role/datazone*",
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
```

```
     }
    },
    {
     "Sid": "AmazonDataZoneCFStackCreationForEnvironments",
     "Effect": "Allow",
     "Action": [
      "cloudformation:CreateStack",
      "cloudformation:TagResource"
     ],
     "Resource": [
      "arn:aws:cloudformation:*:*:stack/DataZone*"
     ],
     "Condition": {
      "ForAnyValue:StringLike": {
       "aws:TagKeys": "AmazonDataZoneEnvironment"
      },
      "Null": {
       "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
      }
     }
    },
    {
     "Sid": "AmazonDataZoneCFStackManagementForEnvironments",
     "Effect": "Allow",
     "Action": [
      "cloudformation:DeleteStack",
      "cloudformation:DescribeStacks",
      "cloudformation:DescribeStackEvents"
     ],
     "Resource": [
      "arn:aws:cloudformation:*:*:stack/DataZone*"
     ]
    },
    {
     "Sid": "AmazonDataZoneEnvironmentParameterValidation",
     "Effect": "Allow",
     "Action": [
      "lakeformation:GetDataLakeSettings",
      "lakeformation:PutDataLakeSettings",
      "lakeformation:RevokePermissions",
      "lakeformation:ListPermissions",
      "glue:CreateDatabase",
      "glue:GetDatabase",
      "athena:GetWorkGroup",
      "logs:DescribeLogGroups",
      "redshift-serverless:GetNamespace",
      "redshift-serverless:GetWorkgroup",
      "redshift:DescribeClusters",
      "secretsmanager:ListSecrets"
     ],
     "Resource": "*"
    },
    {
     "Sid": "AmazonDataZoneEnvironmentLakeFormationPermissions",
     "Effect": "Allow",
     "Action": [
      "lakeformation:RegisterResource",
      "lakeformation:DeregisterResource",
      "lakeformation:GrantPermissions",
      "lakeformation:ListResources"
     ],
     "Resource": "*",
     "Condition": {
      "StringEquals": {
       "aws:CalledViaFirst": [
        "cloudformation.amazonaws.com"
       ]
      }
     }
    },
    {
     "Sid": "AmazonDataZoneEnvironmentGlueDeletePermissions",
     "Effect": "Allow",
     "Action": [
      "glue:DeleteDatabase"
     ],
     "Resource": "*",
     "Condition": {
      "StringEquals": {
       "aws:CalledViaFirst": [
        "cloudformation.amazonaws.com"
       ]
      }
     }
    },
    {
     "Sid": "AmazonDataZoneEnvironmentAthenaDeletePermissions",
     "Effect": "Allow",
     "Action": [
      "athena:DeleteWorkGroup"
     ],
     "Resource": "*",
     "Condition": {
      "StringEquals": {
       "aws:CalledViaFirst": [
        "cloudformation.amazonaws.com"
       ]
      }
     }
    },
    {
     "Sid": "AmazonDataZoneEnvironmentAthenaResourceCreation",
     "Effect": "Allow",
     "Action": [
      "athena:CreateWorkGroup",
      "athena:TagResource",
      "iam:TagRole",
      "iam:TagPolicy",
      "logs:TagLogGroup"
     ],
     "Resource": "*",
     "Condition": {
      "ForAnyValue:StringLike": {
       "aws:TagKeys": "AmazonDataZoneEnvironment"
      },
      "Null": {
       "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
      },
      "StringEquals": {
       "aws:CalledViaFirst": [
        "cloudformation.amazonaws.com"
       ]
      }
     }
    },
    {
     "Sid": "AmazonDataZoneEnvironmentLogGroupCreation",
     "Effect": "Allow",
     "Action": [
      "logs:CreateLogGroup",
      "logs:DeleteLogGroup"
     ],
     "Resource": "arn:aws:logs:*:*:log-group:datazone-*",
     "Condition": {
      "ForAnyValue:StringLike": {
       "aws:TagKeys": "AmazonDataZoneEnvironment"
      },
      "Null": {
       "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
      },
      "StringEquals": {
       "aws:CalledViaFirst": [
        "cloudformation.amazonaws.com"
       ]
      }
     }
    },
    {
     "Sid": "AmazonDataZoneEnvironmentLogGroupManagement",
     "Action": [
      "logs:PutRetentionPolicy"
     ],
     "Resource": "arn:aws:logs:*:*:log-group:datazone-*",
     "Effect": "Allow",
     "Condition": {
      "StringEquals": {
       "aws:CalledViaFirst": [
        "cloudformation.amazonaws.com"
       ]
      }
     }
    },
    {
     "Sid": "AmazonDataZoneEnvironmentIAMPolicyManagement",
     "Effect": "Allow",
     "Action": [
      "iam:DeletePolicy",
      "iam:CreatePolicy",
      "iam:GetPolicy",
      "iam:ListPolicyVersions"
     ],
     "Resource": [
      "arn:aws:iam::*:policy/datazone*"
     ],
     "Condition": {
      "StringEquals": {
```

```
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AmazonDataZoneEnvironmentS3ValidationPermissions",
   "Effect": "Allow",
   "Action": [
    "s3:ListAllMyBuckets",
    "s3:ListBucket"
   ],
   "Resource": "arn:aws:s3:::*"
  },
  {
   "Sid": "AmazonDataZoneEnvironmentKMSDecryptPermissions",
   "Effect": "Allow",
   "Action": [
    "kms:GenerateDataKey",
    "kms:Decrypt"
   ],
   "Resource": "*",
   "Condition": {
    "Null": {
     "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "PermissionsToTagAmazonDataZoneEnvironmentGlueResources",
   "Effect": "Allow",
   "Action": [
    "glue:TagResource"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringLike": {
     "aws:TagKeys": "AmazonDataZoneEnvironment"
    },
    "Null": {
     "aws:RequestTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "PermissionsToGetAmazonDataZoneEnvironmentBlueprintTemplates",
   "Effect": "Allow",
   "Action": "s3:GetObject",
   "Resource": "*",
   "Condition": {
    "StringNotEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    },
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "RedshiftDataPermissions",
   "Effect": "Allow",
   "Action": [
    "redshift-data:ListSchemas",
    "redshift-data:ExecuteStatement"
   ],
   "Resource": [
    "arn:aws:redshift-serverless:*:*:workgroup/*",
    "arn:aws:redshift:*:*:cluster:*"
   ]
  },
  {
   "Sid": "DescribeStatementPermissions",
   "Effect": "Allow",
   "Action": [
    "redshift-data:DescribeStatement"
   ],
   "Resource": "*"
  },
  {
   "Sid": "GetSecretValuePermissions",
   "Effect": "Allow",
   "Action": [
    "secretsmanager:GetSecretValue"
   ],
   "Resource": "*",
   "Condition": {
    "StringLike": {
     "secretsmanager:ResourceTag/AmazonDataZoneDomain": "dzd*"
    }
   }
  }
 ]
}
```

AWS managed policy: AmazonDataZoneGlueManageAccessRolePolicy

This policy gives Amazon DataZone permissions to publish AWS Glue data to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to AWS Glue published assets in the catalog.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "GlueTagDatabasePermissions",
   "Effect": "Allow",
   "Action": [
    "glue:TagResource",
    "glue:UntagResource",
    "glue:GetTags"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    },
    "ForAnyValue:StringLikeIfExists": {
     "aws:TagKeys": "DataZoneDiscoverable_*"
    }
   }
  },
  {
   "Sid": "GlueDataQualityPermissions",
   "Effect": "Allow",
   "Action": [
    "glue:ListDataQualityResults",
    "glue:GetDataQualityResult"
   ],
   "Resource": "arn:aws:glue:*:*:dataQualityRuleset/*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "GlueTableDatabasePermissions",
   "Effect": "Allow",
   "Action": [
    "glue:CreateTable",
    "glue:DeleteTable",
    "glue:GetDatabases",
    "glue:GetTables"
   ],
   "Resource": [
    "arn:aws:glue:*:*:catalog",
    "arn:aws:glue:*:*:database/*",
    "arn:aws:glue:*:*:table/*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "LakeformationResourceSharingPermissions",
   "Effect": "Allow",
   "Action": [
    "lakeformation:BatchGrantPermissions",
    "lakeformation:BatchRevokePermissions",
    "lakeformation:CreateLakeFormationOptIn",
    "lakeformation:DeleteLakeFormationOptIn",
    "lakeformation:GrantPermissions",
    "lakeformation:GetResourceLFTags",
    "lakeformation:ListLakeFormationOptIns",
    "lakeformation:ListPermissions",
    "lakeformation:RegisterResource",
    "lakeformation:RevokePermissions",
    "glue:GetDatabase",
    "glue:GetTable",
    "organizations:DescribeOrganization",
    "ram:GetResourceShareInvitations",
    "ram:ListResources"
   ],
   "Resource": "*"
  },
  {
   "Sid": "CrossAccountRAMResourceSharingPermissions",
   "Effect": "Allow",
   "Action": [
    "glue:DeleteResourcePolicy",
    "glue:PutResourcePolicy"
   ],
   "Resource": [
    "arn:aws:glue:*:*:catalog",
    "arn:aws:glue:*:*:database/*",
    "arn:aws:glue:*:*:table/*"
   ],
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "ram.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "CrossAccountLakeFormationResourceSharingPermissions",
   "Effect": "Allow",
   "Action": [
    "ram:CreateResourceShare"
   ],
   "Resource": "*",
   "Condition": {
    "StringEqualsIfExists": {
     "ram:RequestedResourceType": [
      "glue:Table",
      "glue:Database",
      "glue:Catalog"
     ]
    },
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "lakeformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "CrossAccountRAMResourceShareInvitationPermission",
   "Effect": "Allow",
   "Action": [
    "ram:AcceptResourceShareInvitation"
   ],
   "Resource": "arn:aws:ram:*:*:resource-share-invitation/*"
  },
  {
   "Sid": "CrossAccountRAMResourceSharingViaLakeFormationPermissions",
   "Effect": "Allow",
   "Action": [
    "ram:AssociateResourceShare",
    "ram:DeleteResourceShare",
    "ram:DisassociateResourceShare",
    "ram:GetResourceShares",
    "ram:ListResourceSharePermissions",
    "ram:UpdateResourceShare"
   ],
   "Resource": "*",
   "Condition": {
    "StringLike": {
     "ram:ResourceShareName": [
      "LakeFormation*"
     ]
    },
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "lakeformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "CrossAccountRAMResourceSharingViaLakeFormationHybrid",
   "Effect": "Allow",
   "Action": "ram:AssociateResourceSharePermission",
   "Resource": "*",
   "Condition": {
    "StringLike": {
     "ram:PermissionArn": "arn:aws:ram::aws:permission/AWSRAMLFEnabled*"
    },
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "lakeformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "KMSDecryptPermission",
   "Effect": "Allow",
   "Action": [
    "kms:Decrypt"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceTag/datazone:projectId": "proj-all"
    }
   }
  },
  {
   "Sid": "GetRoleForDataZone",
   "Effect": "Allow",
   "Action": [
    "iam:GetRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/AmazonDataZone*",
    "arn:aws:iam::*:role/service-role/AmazonDataZone*"
   ]
  },
  {
   "Sid": "PassRoleForDataLocationRegistration",
   "Effect": "Allow",
   "Action": [
    "iam:PassRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/AmazonDataZone*",
    "arn:aws:iam::*:role/service-role/AmazonDataZone*"
   ],
   "Condition": {
    "StringEquals": {
     "iam:PassedToService": [
      "lakeformation.amazonaws.com"
     ]
    }
   }
  }
 ]
}
```

AWS managed policy: AmazonDataZoneRedshiftManageAccessRolePolicy

This policy gives Amazon DataZone permissions to publish Amazon Redshift data to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to Amazon Redshift or Amazon Redshift Serverless published assets in the catalog.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "redshiftDataScopeDownPermissions",
   "Effect": "Allow",
   "Action": [
    "redshift-data:BatchExecuteStatement",
    "redshift-data:DescribeTable",
    "redshift-data:ExecuteStatement",
    "redshift-data:ListTables",
    "redshift-data:ListSchemas",
    "redshift-data:ListDatabases"
   ],
   "Resource": [
    "arn:aws:redshift-serverless:*:*:workgroup/*",
    "arn:aws:redshift:*:*:cluster:*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "listSecretsPermission",
   "Effect": "Allow",
   "Action": "secretsmanager:ListSecrets",
   "Resource": "*"
  },
  {
   "Sid": "getWorkgroupPermission",
   "Effect": "Allow",
   "Action": "redshift-serverless:GetWorkgroup",
   "Resource": [
    "arn:aws:redshift-serverless:*:*:workgroup/*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "getNamespacePermission",
   "Effect": "Allow",
   "Action": "redshift-serverless:GetNamespace",
   "Resource": [
    "arn:aws:redshift-serverless:*:*:namespace/*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "redshiftDataPermissions",
   "Effect": "Allow",
   "Action": [
    "redshift-data:DescribeStatement",
    "redshift-data:GetStatementResult",
    "redshift:DescribeClusters"
   ],
   "Resource": "*"
  },
  {
   "Sid": "dataSharesPermissions",
   "Effect": "Allow",
   "Action": [
    "redshift:AuthorizeDataShare",
    "redshift:DescribeDataShares"
   ],
   "Resource": [
    "arn:aws:redshift:*:*:datashare:*/datazone*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "associateDataShareConsumerPermission",
   "Effect": "Allow",
   "Action": "redshift:AssociateDataShareConsumer",
   "Resource": "arn:aws:redshift:*:*:datashare:*/datazone*"
  }
 ]
}
```

AWS managed policy: AmazonDataZoneCrossAccountAdmin

You can attach the AmazonDataZoneCrossAccountAdmin policy to your IAM identities.

This policy enables users to work with Amazon DataZone associated accounts.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ram:UpdateResourceShare",
                "ram:DeleteResourceShare",
                "ram:AssociateResourceShare",
                "ram:DisassociateResourceShare",
                "ram:GetResourceShares"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ram:ResourceShareName": [
                        "DataZone*"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "datazone:PutEnvironmentBlueprintConfiguration",
                "datazone:GetEnvironmentBlueprintConfiguration",
                "datazone:DeleteEnvironmentBlueprintConfiguration",
                "datazone:ListEnvironmentBlueprintConfigurations",
                "datazone:ListDomains",
                "datazone:GetDomain",
                "datazone:GetEnvironmentBlueprint",
                "datazone:ListEnvironmentBlueprints",
                "datazone:ListEnvironments",
                "datazone:GetEnvironment",
                "ram:AcceptResourceShareInvitation",
                "ram:RejectResourceShareInvitation",
                "ram:Get*",
                "ram:List*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AmazonDataZoneDomainExecutionRolePolicy

This is the default policy for the Amazon DataZone `DomainExecutionRole` service role. This role is used by Amazon DataZone to catalog, discover, govern, share, and analyze data in the Amazon DataZone domain. This role provides access to all Amazon DataZone APIs that are required for data portal use, as well as RAM permissions to support usage of associated accounts in a Amazon DataZone domain.

You can attach the AmazonDataZoneDomainExecutionRolePolicy policy to your `AmazonDataZoneDomainExecutionRole`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DomainExecutionRoleStatement",
            "Effect": "Allow",
            "Action": [
                "datazone:ListTimeSeriesDataPoints",
                "datazone:GetTimeSeriesDataPoint",
                "datazone:DeleteTimeSeriesDataPoints",
                "datazone:AcceptPredictions",
                "datazone:AcceptSubscriptionRequest",
                "datazone:CancelSubscription",
                "datazone:CreateAsset",
                "datazone:CreateAssetRevision",
                "datazone:CreateAssetType",
                "datazone:CreateDataSource",
                "datazone:CreateEnvironment",
                "datazone:CreateEnvironmentBlueprint",
                "datazone:CreateEnvironmentProfile",
                "datazone:CreateFormType",
                "datazone:CreateGlossary",
                "datazone:CreateGlossaryTerm",
                "datazone:CreateListingChangeSet",
                "datazone:CreateProject",
                "datazone:CreateProjectMembership",
                "datazone:CreateSubscriptionGrant",
                "datazone:CreateSubscriptionRequest",
                "datazone:DeleteAsset",
                "datazone:DeleteAssetType",
                "datazone:DeleteDataSource",
                "datazone:DeleteEnvironment",
                "datazone:DeleteEnvironmentBlueprint",
                "datazone:DeleteEnvironmentProfile",
                "datazone:DeleteFormType",
                "datazone:DeleteGlossary",
                "datazone:DeleteGlossaryTerm",
                "datazone:DeleteListing",
                "datazone:DeleteProject",
                "datazone:DeleteProjectMembership",
                "datazone:DeleteSubscriptionGrant",
                "datazone:DeleteSubscriptionRequest",
                "datazone:DeleteSubscriptionTarget",
                "datazone:GetAsset",
                "datazone:GetAssetType",
                "datazone:GetDataSource",
                "datazone:GetDataSourceRun",
                "datazone:GetDomain",
                "datazone:GetEnvironment",
                "datazone:GetEnvironmentActionLink",
                "datazone:GetEnvironmentBlueprint",
                "datazone:GetEnvironmentCredentials",
                "datazone:GetEnvironmentProfile",
                "datazone:GetFormType",
                "datazone:GetGlossary",
                "datazone:GetGlossaryTerm",
                "datazone:GetGroupProfile",
                "datazone:GetListing",
                "datazone:GetProject",
                "datazone:GetSubscription",
                "datazone:GetSubscriptionEligibility",
                "datazone:GetSubscriptionGrant",
                "datazone:GetSubscriptionRequestDetails",
                "datazone:GetSubscriptionTarget",
                "datazone:GetUserProfile",
                "datazone:ListAccountEnvironments",
                "datazone:ListAssetRevisions",
                "datazone:ListDataSourceRunActivities",
                "datazone:ListDataSourceRuns",
                "datazone:ListDataSources",
                "datazone:ListEnvironmentBlueprintConfigurations",
                "datazone:ListEnvironmentBlueprintConfigurationSummaries",
                "datazone:ListEnvironmentBlueprints",
                "datazone:ListEnvironmentProfiles",
                "datazone:ListEnvironments",
                "datazone:ListGroupsForUser",
                "datazone:ListNotifications",
                "datazone:ListProjectMemberships",
                "datazone:ListProjects",
                "datazone:ListSubscriptionGrants",
                "datazone:ListSubscriptionRequests",
                "datazone:ListSubscriptionTargets",
                "datazone:ListSubscriptions",
                "datazone:ListWarehouseMetadata",
                "datazone:RejectPredictions",
                "datazone:RejectSubscriptionRequest",
                "datazone:RevokeSubscription",
                "datazone:Search",
                "datazone:SearchGroupProfiles",
                "datazone:SearchListings",
                "datazone:SearchTypes",
                "datazone:SearchUserProfiles",
                "datazone:StartDataSourceRun",
                "datazone:UpdateDataSource",
                "datazone:UpdateEnvironment",
                "datazone:UpdateEnvironmentBlueprint",
                "datazone:UpdateEnvironmentDeploymentStatus",
                "datazone:UpdateEnvironmentProfile",
                "datazone:UpdateGlossary",
                "datazone:UpdateGlossaryTerm",
                "datazone:UpdateProject",
                "datazone:UpdateSubscriptionGrantStatus",
                "datazone:UpdateSubscriptionRequest",
                "datazone:StartMetadataGenerationRun",
                "datazone:GetMetadataGenerationRun",
                "datazone:CancelMetadataGenerationRun",
                "datazone:ListMetadataGenerationRuns",
                "datazone:GetEnvironmentAction",
                "datazone:ListEnvironmentActions",
                "datazone:ListLineageNodeHistory",
                "datazone:GetLineageNode",
                "datazone:CreateAssetFilter",
                "datazone:DeleteAssetFilter",
                "datazone:GetAssetFilter",
                "datazone:ListAssetFilters",
                "datazone:UpdateAssetFilter"
            ],
            "Resource": "*"
        },
        {
            "Sid": "RAMResourceShareStatement",
            "Effect": "Allow",
            "Action": "ram:GetResourceShareAssociations",
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AmazonDataZoneSageMakerProvisioning

The AmazonDataZoneSageMakerProvisioning policy grants Amazon DataZone the permissions required to interoperate with Amazon SageMaker.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "CreateSageMakerStudio",
   "Effect": "Allow",
   "Action": [
    "sagemaker:CreateDomain"
   ],
   "Resource": [
    "*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    },
    "ForAnyValue:StringEquals": {
     "aws:TagKeys": [
      "AmazonDataZoneEnvironment"
     ]
    },
    "Null": {
     "aws:TagKeys": "false",
     "aws:ResourceTag/AmazonDataZoneEnvironment": "false",
     "aws:RequestTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "DeleteSageMakerStudio",
   "Effect": "Allow",
   "Action": [
    "sagemaker:DeleteDomain"
   ],
   "Resource": [
    "*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    },
    "ForAnyValue:StringLike": {
     "aws:TagKeys": [
      "AmazonDataZoneEnvironment"
     ]
    },
    "Null": {
     "aws:TagKeys": "false",
     "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "AmazonDataZoneEnvironmentSageMakerDescribePermissions",
   "Effect": "Allow",
   "Action": [
    "sagemaker:DescribeDomain"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "IamPassRolePermissions",
   "Effect": "Allow",
   "Action": [
    "iam:PassRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/sm-provisioning/datazone_usr*"
   ],
   "Condition": {
    "StringEquals": {
     "iam:PassedToService": [
      "glue.amazonaws.com",
      "lakeformation.amazonaws.com",
      "sagemaker.amazonaws.com"
     ],
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AmazonDataZonePermissionsToCreateEnvironmentRole",
   "Effect": "Allow",
   "Action": [
    "iam:CreateRole",
    "iam:DetachRolePolicy",
    "iam:DeleteRolePolicy",
    "iam:AttachRolePolicy",
    "iam:PutRolePolicy"
   ],
   "Resource": [
    "arn:aws:iam::*:role/sm-provisioning/datazone_usr*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ],
     "iam:PermissionsBoundary": "arn:aws:iam::aws:policy/
AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary"
    }
   }
  },
  {
   "Sid": "AmazonDataZonePermissionsToManageEnvironmentRole",
   "Effect": "Allow",
   "Action": [
    "iam:GetRole",
    "iam:GetRolePolicy",
    "iam:DeleteRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/sm-provisioning/datazone_usr*"
   ],
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AmazonDataZonePermissionsToCreateSageMakerServiceRole",
   "Effect": "Allow",
   "Action": [
    "iam:CreateServiceLinkedRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/aws-service-role/sagemaker.amazonaws.com/
AWSServiceRoleForAmazonSageMakerNotebooks"
   ],
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AmazonDataZoneEnvironmentParameterValidation",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeVpcs",
    "ec2:DescribeSubnets",
    "sagemaker:ListDomains"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AmazonDataZoneEnvironmentKMSKeyValidation",
   "Effect": "Allow",
   "Action": [
    "kms:DescribeKey"
   ],
   "Resource": "arn:aws:kms:*:*:key/*",
   "Condition": {
    "Null": {
     "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "AmazonDataZoneEnvironmentGluePermissions",
   "Effect": "Allow",
   "Action": [
    "glue:CreateConnection",
    "glue:DeleteConnection"
   ],
   "Resource": [
    "arn:aws:glue:*:*:connection/dz-sm-athena-glue-connection-*",
    "arn:aws:glue:*:*:connection/dz-sm-redshift-cluster-connection-*",
    "arn:aws:glue:*:*:connection/dz-sm-redshift-serverless-connection-*",
    "arn:aws:glue:*:*:catalog"
   ],
   "Condition": {
    "StringEquals": {
     "aws:CalledViaFirst": [
      "cloudformation.amazonaws.com"
     ]
    }
   }
  }
 ]
}
```

AWS managed policy: AmazonDataZoneSageMakerAccess

This policy gives Amazon DataZone permissions to publish Amazon SageMaker assets to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to the Amazon SageMaker published assets in the catalog.

This policy includes permissions to do the following:

- cloudtrail – retrieve information about CloudTrail trails.

- cloudwatch – retrieve the current CloudWatch alarms.
- logs – retrieve the metric filters for CloudWatch logs.
- sns – retrieve the list of subscriptions to an SNS topic.
- config – retrieve information about configuration recorders, resources, and AWS Config rules. Also allows the service-linked role to create and delete AWS Config rules, and to run evaluations against the rules.
- iam – get and generate credential reports for accounts.
- organizations – retrieve account and organizational unit (OU) information for an organization.
- securityhub – retrieve information about how the Security Hub service, standards, and controls are configured.
- tag – retrieve information about resource tags.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AmazonSageMakerReadPermission",
   "Effect": "Allow",
   "Action": [
    "sagemaker:DescribeFeatureGroup",
    "sagemaker:ListModelPackages",
    "sagemaker:DescribeModelPackage",
    "sagemaker:DescribeModelPackageGroup",
    "sagemaker:DescribeAlgorithm",
    "sagemaker:ListTags",
    "sagemaker:DescribeDomain",
    "sagemaker:GetModelPackageGroupPolicy",
    "sagemaker:Search"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AmazonSageMakerTaggingPermission",
   "Effect": "Allow",
   "Action": [
    "sagemaker:AddTags",
    "sagemaker:DeleteTags"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringLike": {
     "aws:TagKeys": [
      "sagemaker:shared-with:*"
     ]
    }
   }
  },
  {
   "Sid": "AmazonSageMakerModelPackageGroupPolicyPermission",
   "Effect": "Allow",
   "Action": [
    "sagemaker:PutModelPackageGroupPolicy",
    "sagemaker:DeleteModelPackageGroupPolicy"
   ],
   "Resource": [
    "arn:*:sagemaker:*:*:model-package-group/*"
   ]
  },
  {
   "Sid": "AmazonSageMakerRAMPermission",
   "Effect": "Allow",
   "Action": [
    "ram:GetResourceShares",
    "ram:GetResourceShareInvitations",
    "ram:GetResourceShareAssociations"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AmazonSageMakerRAMResourcePolicyPermission",
   "Effect": "Allow",
   "Action": [
    "sagemaker:PutResourcePolicy",
    "sagemaker:GetResourcePolicy",
    "sagemaker:DeleteResourcePolicy"
   ],
   "Resource": [
    "arn:*:sagemaker:*:*:feature-group/*"
   ]
  },
  {
   "Sid": "AmazonSageMakerRAMTagResourceSharePermission",
   "Effect": "Allow",
   "Action": [
    "ram:TagResource"
   ],
   "Resource": "arn:*:ram:*:*:resource-share/*",
   "Condition": {
    "Null": {
     "aws:RequestTag/AwsDataZoneDomainId": "false"
    }
   }
  },
  {
   "Sid": "AmazonSageMakerRAMDeleteResourceSharePermission",
   "Effect": "Allow",
   "Action": [
    "ram:DeleteResourceShare"
   ],
   "Resource": "arn:*:ram:*:*:resource-share/*",
   "Condition": {
    "Null": {
     "aws:ResourceTag/AwsDataZoneDomainId": "false"
    }
   }
  },
  {
   "Sid": "AmazonSageMakerRAMCreateResourceSharePermission",
   "Effect": "Allow",
   "Action": [
    "ram:CreateResourceShare"
   ],
   "Resource": "*",
   "Condition": {
    "StringLikeIfExists": {
     "ram:RequestedResourceType": [
      "sagemaker:*"
     ]
    },
    "Null": {
     "aws:RequestTag/AwsDataZoneDomainId": "false"
    }
   }
  },
  {
   "Sid": "AmazonSageMakerS3BucketPolicyPermission",
   "Effect": "Allow",
   "Action": [
    "s3:DeleteBucketPolicy",
    "s3:PutBucketPolicy",
    "s3:GetBucketPolicy"
   ],
   "Resource": [
    "arn:aws:s3:::sagemaker-datazone*",
    "arn:aws:s3:::SageMaker-DataZone*",
    "arn:aws:s3:::datazone-sagemaker*",
    "arn:aws:s3:::DataZone-SageMaker*",
    "arn:aws:s3:::amazon-datazone*"
   ]
  },
  {
   "Sid": "AmazonSageMakerS3Permission",
   "Effect": "Allow",
   "Action": [
    "s3:GetObject",
    "s3:ListBucket"
   ],
   "Resource": [
    "arn:aws:s3:::sagemaker-datazone*",
    "arn:aws:s3:::SageMaker-DataZone*",
    "arn:aws:s3:::datazone-sagemaker*",
    "arn:aws:s3:::DataZone-SageMaker*",
    "arn:aws:s3:::amazon-datazone*"
   ]
  },
  {
   "Sid": "AmazonSageMakerECRPermission",
   "Effect": "Allow",
   "Action": [
    "ecr:GetRepositoryPolicy",
    "ecr:SetRepositoryPolicy",
    "ecr:DeleteRepositoryPolicy"
   ],
   "Resource": "*",
   "Condition": {
    "Null": {
     "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "AmazonSageMakerKMSReadPermission",
   "Effect": "Allow",
   "Action": [
    "kms:DescribeKey"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:TagKeys": [
      "AmazonDataZoneEnvironment"
     ]
    }
   }
  },
  {
   "Sid": "AmazonSageMakerKMSGrantPermission",
   "Effect": "Allow",
   "Action": [
    "kms:CreateGrant"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:TagKeys": [
      "AmazonDataZoneEnvironment"
     ]
    },
    "ForAllValues:StringEquals": {
     "kms:GrantOperations": [
      "Decrypt"
     ]
    }
   }
  }
 ]
}
```

AWS managed policy: AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary

> **Note**
>
> This policy is a *permissions boundary*. A permissions boundary sets the maximum permissions that an identity-based policy can grant to an IAM entity. You should not use and attach Amazon DataZone permissions boundary policies on your own. Amazon DataZone permissions boundary policies should only be attached to Amazon DataZone managed roles. For more information on permissions boundaries, see Permissions boundaries for IAM entities in the IAM User Guide.

When you create an Amazon SageMaker environment via the Amazon DataZone data portal, Amazon DataZone applies this permissions boundary to the IAM roles that are produced during environment creation. The permissions boundary limits the scope of the roles that Amazon DataZone creates and any roles that you add.

Amazon DataZone uses the `AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary` managed policy to limit the provisioned IAM principal to which it is attached. The principals might take the form of the user roles that Amazon DataZone can assume on behalf of interactive enterprise users or analytic services (AWS SageMaker, for example), and then conduct actions to process data such as reading and writing from Amazon S3 or Amazon Redshift or running AWS Glue crawler.

The `AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary` policy grants read and write access for Amazon DataZone to services such as Amazon SageMaker, AWS Glue, Amazon S3, AWS Lake Formation, Amazon Redshift, and Amazon Athena. The policy also gives read and write permissions to some infrastructure resources that are required to use these services such as network interfaces, Amazon ECR repositories and AWS KMS keys. It also give access to Amazon SageMaker applications like Amazon SageMaker Canvas.

Amazon DataZone applies the `AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary` managed policy as a permissions boundary for all Amazon DataZone environment roles (owner and contributor). This permissions boundary restricts these roles to only allow access to the required resources and actions necessary for an environment.

```
       {
"Version": "2012-10-17",
"Statement": [
 {
  "Sid": "AllowAllNonAdminSageMakerActions",
  "Effect": "Allow",
  "Action": [
   "sagemaker:*",
   "sagemaker-geospatial:*"
  ],
  "NotResource": [
   "arn:aws:sagemaker:*:*:domain/*",
   "arn:aws:sagemaker:*:*:user-profile/*",
   "arn:aws:sagemaker:*:*:app/*",
   "arn:aws:sagemaker:*:*:space/*",
   "arn:aws:sagemaker:*:*:flow-definition/*"
  ]
 },
 {
  "Sid": "AllowSageMakerProfileManagement",
  "Effect": "Allow",
  "Action": [
   "sagemaker:CreateUserProfile",
   "sagemaker:DescribeUserProfile",
   "sagemaker:UpdateUserProfile",
   "sagemaker:CreatePresignedDomainUrl"
  ],
  "Resource": "arn:aws:sagemaker:*:*:*/*"
 },
 {
  "Sid": "AllowLakeFormation",
  "Effect": "Allow",
  "Action": [
   "lakeformation:GetDataAccess"
  ],
  "Resource": "*"
 },
 {
  "Sid": "AllowAddTagsForAppAndSpace",
  "Effect": "Allow",
  "Action": [
   "sagemaker:AddTags"
  ],
  "Resource": [
   "arn:aws:sagemaker:*:*:app/*",
   "arn:aws:sagemaker:*:*:space/*"
  ],
  "Condition": {
   "StringEquals": {
    "sagemaker:TaggingAction": [
     "CreateApp",
     "CreateSpace"
    ]
   }
  }
 },
 {
  "Sid": "AllowStudioActions",
  "Effect": "Allow",
  "Action": [
   "sagemaker:CreatePresignedDomainUrl",
   "sagemaker:DescribeApp",
   "sagemaker:DescribeDomain",
   "sagemaker:DescribeSpace",
   "sagemaker:DescribeUserProfile",
   "sagemaker:ListApps",
   "sagemaker:ListDomains",
   "sagemaker:ListSpaces",
   "sagemaker:ListUserProfiles"
  ],
  "Resource": "*"
 },
 {
  "Sid": "AllowAppActionsForUserProfile",
  "Effect": "Allow",
  "Action": [
   "sagemaker:CreateApp",
   "sagemaker:DeleteApp"
  ],
  "Resource": "arn:aws:sagemaker:*:*:app/*/*/*/*",
  "Condition": {
   "Null": {
    "sagemaker:OwnerUserProfileArn": "true"
   }
  }
 },
 {
  "Sid": "AllowAppActionsForSharedSpaces",
  "Effect": "Allow",
  "Action": [
   "sagemaker:CreateApp",
   "sagemaker:DeleteApp"
  ],
  "Resource": "arn:aws:sagemaker:*:*:app/${sagemaker:DomainId}/*/*/*",
  "Condition": {
   "StringEquals": {
    "sagemaker:SpaceSharingType": [
     "Shared"
    ]
   }
  }
 },
 {
  "Sid": "AllowMutatingActionsOnSharedSpacesWithoutOwner",
  "Effect": "Allow",
  "Action": [
   "sagemaker:CreateSpace",
   "sagemaker:DeleteSpace",
   "sagemaker:UpdateSpace"
  ],
  "Resource": "arn:aws:sagemaker:*:*:space/${sagemaker:DomainId}/*",
  "Condition": {
   "Null": {
    "sagemaker:OwnerUserProfileArn": "true"
   }
  }
 },
 {
  "Sid": "RestrictMutatingActionsOnSpacesToOwnerUserProfile",
  "Effect": "Allow",
  "Action": [
   "sagemaker:CreateSpace",
   "sagemaker:DeleteSpace",
   "sagemaker:UpdateSpace"
  ],
  "Resource": "arn:aws:sagemaker:*:*:space/${sagemaker:DomainId}/*",
  "Condition": {
   "ArnLike": {
    "sagemaker:OwnerUserProfileArn": "arn:aws:sagemaker:*:*:user-profile/
${sagemaker:DomainId}/${sagemaker:UserProfileName}"
```

```
    },
    "StringEquals": {
     "sagemaker:SpaceSharingType": [
      "Private",
      "Shared"
     ]
    }
   }
  },
  {
   "Sid": "RestrictMutatingActionsOnPrivateSpaceAppsToOwnerUserProfile",
   "Effect": "Allow",
   "Action": [
    "sagemaker:CreateApp",
    "sagemaker:DeleteApp"
   ],
   "Resource": "arn:aws:sagemaker:*:*:app/${sagemaker:DomainId}/*/*/*",
   "Condition": {
    "ArnLike": {
     "sagemaker:OwnerUserProfileArn": "arn:aws:sagemaker:*:*:user-profile/
${sagemaker:DomainId}/${sagemaker:UserProfileName}"
    },
    "StringEquals": {
     "sagemaker:SpaceSharingType": [
      "Private"
     ]
    }
   }
  },
  {
   "Sid": "AllowFlowDefinitionActions",
   "Effect": "Allow",
   "Action": "sagemaker:*",
   "Resource": [
    "arn:aws:sagemaker:*:*:flow-definition/*"
   ],
   "Condition": {
    "StringEqualsIfExists": {
     "sagemaker:WorkteamType": [
      "private-crowd",
      "vendor-crowd"
     ]
    }
   }
  },
  {
   "Sid": "AllowAWSServiceActions",
   "Effect": "Allow",
   "Action": [
    "sqlworkbench:*",
    "datazone:*",
    "application-autoscaling:DeleteScalingPolicy",
    "application-autoscaling:DeleteScheduledAction",
    "application-autoscaling:DeregisterScalableTarget",
    "application-autoscaling:DescribeScalableTargets",
    "application-autoscaling:DescribeScalingActivities",
    "application-autoscaling:DescribeScalingPolicies",
    "application-autoscaling:DescribeScheduledActions",
    "application-autoscaling:PutScalingPolicy",
    "application-autoscaling:PutScheduledAction",
    "application-autoscaling:RegisterScalableTarget",
    "aws-marketplace:ViewSubscriptions",
    "cloudformation:GetTemplateSummary",
    "cloudwatch:DeleteAlarms",
    "cloudwatch:DescribeAlarms",
    "cloudwatch:GetMetricData",
    "cloudwatch:GetMetricStatistics",
    "cloudwatch:ListMetrics",
    "cloudwatch:PutMetricAlarm",
    "cloudwatch:PutMetricData",
    "codecommit:BatchGetRepositories",
    "codecommit:CreateRepository",
    "codecommit:GetRepository",
    "codecommit:List*",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DescribeDhcpOptions",
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribeRouteTables",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSubnets",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeVpcEndpointServices",
    "ec2:DescribeVpcs",
    "ecr:BatchCheckLayerAvailability",
    "ecr:BatchGetImage",
    "ecr:Describe*",
    "ecr:GetAuthorizationToken",
    "ecr:GetDownloadUrlForLayer",
    "ecr:StartImageScan",
    "elastic-inference:Connect",
    "elasticfilesystem:DescribeFileSystems",
    "elasticfilesystem:DescribeMountTargets",
    "fsx:DescribeFileSystems",
    "groundtruthlabeling:*",
    "iam:GetRole",
    "iam:ListRoles",
    "kms:DescribeKey",
    "kms:ListAliases",
    "lambda:ListFunctions",
    "logs:CreateLogDelivery",
    "logs:CreateLogGroup",
    "logs:CreateLogStream",
    "logs:DeleteLogDelivery",
    "logs:DescribeLogGroups",
    "logs:DescribeLogStreams",
    "logs:GetLogDelivery",
    "logs:GetLogEvents",
    "logs:ListLogDeliveries",
    "logs:PutLogEvents",
    "logs:UpdateLogDelivery",
    "redshift-data:BatchExecuteStatement",
    "redshift-data:CancelStatement",
    "redshift-data:DescribeStatement",
    "redshift-data:DescribeTable",
    "redshift-data:ExecuteStatement",
    "redshift-data:GetStatementResult",
    "redshift-data:ListSchemas",
    "redshift-data:ListTables",
    "redshift-serverless:GetCredentials",
    "redshift-serverless:GetNamespace",
    "redshift-serverless:GetWorkgroup",
    "redshift-serverless:ListNamespaces",
    "redshift-serverless:ListWorkgroups",
    "secretsmanager:ListSecrets",
    "servicecatalog:Describe*",
    "servicecatalog:List*",
    "servicecatalog:ScanProvisionedProducts",
    "servicecatalog:SearchProducts",
    "servicecatalog:SearchProvisionedProducts",
    "sns:ListTopics",
    "tag:GetResources"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowRAMInvitation",
   "Effect": "Allow",
   "Action": "ram:AcceptResourceShareInvitation",
   "Resource": "*",
   "Condition": {
    "StringLike": {
     "ram:ResourceShareName": "dzd_*"
    }
   }
  },
  {
   "Sid": "AllowECRActions",
   "Effect": "Allow",
   "Action": [
    "ecr:SetRepositoryPolicy",
    "ecr:CompleteLayerUpload",
    "ecr:CreateRepository",
    "ecr:BatchDeleteImage",
    "ecr:UploadLayerPart",
    "ecr:DeleteRepositoryPolicy",
    "ecr:InitiateLayerUpload",
    "ecr:DeleteRepository",
    "ecr:PutImage",
    "ecr:TagResource",
    "ecr:UntagResource"
   ],
   "Resource": [
    "arn:aws:ecr:*:*:repository/sagemaker*",
    "arn:aws:ecr:*:*:repository/datazone*"
   ]
  },
  {
   "Sid": "AllowCodeCommitActions",
   "Effect": "Allow",
   "Action": [
    "codecommit:GitPull",
    "codecommit:GitPush"
   ],
   "Resource": [
    "arn:aws:codecommit:*:*:*sagemaker*",
    "arn:aws:codecommit:*:*:*SageMaker*",
    "arn:aws:codecommit:*:*:*Sagemaker*"
   ]
  },
  {
   "Sid": "AllowCodeBuildActions",
   "Action": [
    "codebuild:BatchGetBuilds",
    "codebuild:StartBuild"
   ],
   "Resource": [
    "arn:aws:codebuild:*:*:project/sagemaker*",
    "arn:aws:codebuild:*:*:build/*"
   ],
   "Effect": "Allow"
  },
  {
   "Sid": "AllowStepFunctionsActions",
   "Action": [
    "states:DescribeExecution",
    "states:GetExecutionHistory",
    "states:StartExecution",
    "states:StopExecution",
    "states:UpdateStateMachine"
   ],
   "Resource": [
    "arn:aws:states:*:*:statemachine:*sagemaker*",
    "arn:aws:states:*:*:execution:*sagemaker*:*"
   ],
   "Effect": "Allow"
  },
  {
   "Sid": "AllowSecretManagerActions",
   "Effect": "Allow",
   "Action": [
    "secretsmanager:DescribeSecret",
    "secretsmanager:GetSecretValue",
    "secretsmanager:CreateSecret",
    "secretsmanager:PutResourcePolicy"
   ],
   "Resource": [
    "arn:aws:secretsmanager:*:*:secret:AmazonSageMaker-*"
```

```
   ]
  },
  {
   "Sid": "AllowServiceCatalogProvisionProduct",
   "Effect": "Allow",
   "Action": [
    "servicecatalog:ProvisionProduct"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowServiceCatalogTerminateUpdateProvisionProduct",
   "Effect": "Allow",
   "Action": [
    "servicecatalog:TerminateProvisionedProduct",
    "servicecatalog:UpdateProvisionedProduct"
   ],
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "servicecatalog:userLevel": "self"
    }
   }
  },
  {
   "Sid": "AllowS3ObjectActions",
   "Effect": "Allow",
   "Action": [
    "s3:AbortMultipartUpload",
    "s3:DeleteObject",
    "s3:DeleteObjectVersion",
    "s3:GetObject",
    "s3:PutObject",
    "s3:PutObjectRetention",
    "s3:ReplicateObject",
    "s3:RestoreObject",
    "s3:GetBucketAcl",
    "s3:PutObjectAcl"
   ],
   "Resource": [
    "arn:aws:s3:::SageMaker-DataZone*",
    "arn:aws:s3:::DataZone-SageMaker*",
    "arn:aws:s3:::Sagemaker-DataZone*",
    "arn:aws:s3:::DataZone-Sagemaker*",
    "arn:aws:s3:::sagemaker-datazone*",
    "arn:aws:s3:::datazone-sagemaker*",
    "arn:aws:s3:::amazon-datazone*"
   ]
  },
  {
   "Sid": "AllowS3GetObjectWithSageMakerExistingObjectTag",
   "Effect": "Allow",
   "Action": [
    "s3:GetObject"
   ],
   "Resource": [
    "arn:aws:s3:::*"
   ],
   "Condition": {
    "StringEqualsIgnoreCase": {
     "s3:ExistingObjectTag/SageMaker": "true"
    }
   }
  },
  {
   "Sid": "AllowS3GetObjectWithServiceCatalogProvisioningExistingObjectTag",
   "Effect": "Allow",
   "Action": [
    "s3:GetObject"
   ],
   "Resource": [
    "arn:aws:s3:::*"
   ],
   "Condition": {
    "StringEquals": {
     "s3:ExistingObjectTag/servicecatalog:provisioning": "true"
    }
   }
  },
  {
   "Sid": "AllowS3BucketActions",
   "Effect": "Allow",
   "Action": [
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:ListAllMyBuckets",
    "s3:GetBucketCors",
    "s3:PutBucketCors"
   ],
   "Resource": [
    "arn:aws:s3:::SageMaker-DataZone*",
    "arn:aws:s3:::DataZone-SageMaker*",
    "arn:aws:s3:::Sagemaker-DataZone*",
    "arn:aws:s3:::DataZone-Sagemaker*",
    "arn:aws:s3:::sagemaker-datazone*",
    "arn:aws:s3:::datazone-sagemaker*",
    "arn:aws:s3:::amazon-datazone*"
   ]
  },
  {
   "Sid": "ReadSageMakerJumpstartArtifacts",
   "Effect": "Allow",
   "Action": "s3:GetObject",
   "Resource": [
    "arn:aws:s3:::jumpstart-cache-prod-us-west-2/*",
    "arn:aws:s3:::jumpstart-cache-prod-us-east-1/*",
    "arn:aws:s3:::jumpstart-cache-prod-us-east-2/*",
    "arn:aws:s3:::jumpstart-cache-prod-eu-west-1/*",
    "arn:aws:s3:::jumpstart-cache-prod-eu-central-1/*",
    "arn:aws:s3:::jumpstart-cache-prod-ap-south-1/*",
    "arn:aws:s3:::jumpstart-cache-prod-ap-northeast-2/*",
    "arn:aws:s3:::jumpstart-cache-prod-ap-northeast-1/*",
    "arn:aws:s3:::jumpstart-cache-prod-ap-southeast-1/*",
    "arn:aws:s3:::jumpstart-cache-prod-ap-southeast-2/*"
   ]
  },
  {
   "Sid": "AllowLambdaInvokeFunction",
   "Effect": "Allow",
   "Action": [
    "lambda:InvokeFunction"
   ],
   "Resource": [
    "arn:aws:lambda:*:*:function:*SageMaker*",
    "arn:aws:lambda:*:*:function:*sagemaker*",
    "arn:aws:lambda:*:*:function:*Sagemaker*",
    "arn:aws:lambda:*:*:function:*LabelingFunction*"
   ]
  },
  {
   "Sid": "AllowCreateServiceLinkedRoleForSageMakerApplicationAutoscaling",
   "Action": "iam:CreateServiceLinkedRole",
   "Effect": "Allow",
   "Resource": "arn:aws:iam::*:role/aws-service-role/sagemaker.application-
autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_SageMakerEndpoint",
   "Condition": {
    "StringLike": {
     "iam:AWSServiceName": "sagemaker.application-autoscaling.amazonaws.com"
    }
   }
  },
  {
   "Sid": "AllowSNSActions",
   "Effect": "Allow",
   "Action": [
    "sns:Subscribe",
    "sns:CreateTopic",
    "sns:Publish"
   ],
   "Resource": [
    "arn:aws:sns:*:*:*SageMaker*",
    "arn:aws:sns:*:*:*Sagemaker*",
    "arn:aws:sns:*:*:*sagemaker*"
   ]
  },
  {
   "Sid": "AllowPassRoleForSageMakerRoles",
   "Effect": "Allow",
   "Action": [
    "iam:PassRole"
   ],
   "Resource": [
    "arn:aws:iam::*:role/sm-provisioning/datazone_usr_sagemaker_execution_role_*"
   ],
   "Condition": {
    "StringEquals": {
     "iam:PassedToService": [
      "glue.amazonaws.com",
      "bedrock.amazonaws.com",
      "states.amazonaws.com",
      "lakeformation.amazonaws.com",
      "events.amazonaws.com",
      "sagemaker.amazonaws.com",
      "forecast.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "CrossAccountKmsOperations",
   "Effect": "Allow",
   "Action": [
    "kms:DescribeKey",
    "kms:Decrypt",
    "kms:ListKeys"
   ],
   "Resource": "*",
   "Condition": {
    "StringNotEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    }
   }
  },
  {
   "Sid": "KmsOperationsWithResourceTag",
   "Effect": "Allow",
   "Action": [
    "kms:DescribeKey",
    "kms:Decrypt",
    "kms:ListKeys",
    "kms:Encrypt",
    "kms:GenerateDataKey",
    "kms:RetireGrant"
   ],
   "Resource": "*",
   "Condition": {
    "Null": {
     "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "AllowAthenaActions",
   "Effect": "Allow",
   "Action": [
    "athena:BatchGetNamedQuery",
    "athena:BatchGetPreparedStatement",
    "athena:BatchGetQueryExecution",
    "athena:CreateNamedQuery",
    "athena:CreateNotebook",
    "athena:CreatePreparedStatement",
    "athena:CreatePresignedNotebookUrl",
    "athena:DeleteNamedQuery",
    "athena:DeleteNotebook",
    "athena:DeletePreparedStatement",
    "athena:ExportNotebook",
    "athena:GetDatabase",
    "athena:GetDataCatalog",
    "athena:GetNamedQuery",
    "athena:GetPreparedStatement",
    "athena:GetQueryExecution",
    "athena:GetQueryResults",
    "athena:GetQueryResultsStream",
    "athena:GetQueryRuntimeStatistics",
    "athena:GetTableMetadata",
    "athena:GetWorkGroup",
    "athena:ImportNotebook",
    "athena:ListDatabases",
    "athena:ListDataCatalogs",
    "athena:ListEngineVersions",
    "athena:ListNamedQueries",
    "athena:ListPreparedStatements",
    "athena:ListQueryExecutions",
    "athena:ListTableMetadata",
    "athena:ListTagsForResource",
    "athena:ListWorkGroups",
    "athena:StartCalculationExecution",
    "athena:StartQueryExecution",
    "athena:StartSession",
    "athena:StopCalculationExecution",
    "athena:StopQueryExecution",
    "athena:TerminateSession",
    "athena:UpdateNamedQuery",
    "athena:UpdateNotebook",
    "athena:UpdateNotebookMetadata",
    "athena:UpdatePreparedStatement"
   ],
   "Resource": [
    "*"
   ]
  },
  {
   "Sid": "AllowGlueCreateDatabase",
   "Effect": "Allow",
   "Action": [
    "glue:CreateDatabase"
   ],
   "Resource": [
    "arn:aws:glue:*:*:catalog",
    "arn:aws:glue:*:*:database/default"
   ]
  },
  {
   "Sid": "AllowRedshiftGetClusterCredentials",
   "Effect": "Allow",
   "Action": [
    "redshift:GetClusterCredentials"
   ],
   "Resource": [
    "arn:aws:redshift:*:*:dbuser:*/sagemaker_access*",
    "arn:aws:redshift:*:*:dbname:*"
   ]
  },
  {
   "Sid": "AllowListTags",
   "Effect": "Allow",
   "Action": [
    "sagemaker:ListTags"
   ],
   "Resource": [
    "arn:aws:sagemaker:*:*:user-profile/*",
    "arn:aws:sagemaker:*:*:domain/*"
   ]
  },
  {
   "Sid": "AllowCloudformationListStackResources",
   "Effect": "Allow",
   "Action": [
    "cloudformation:ListStackResources"
   ],
   "Resource": "arn:aws:cloudformation:*:*:stack/SC-*"
  },
  {
   "Sid": "AllowGlueActions",
   "Effect": "Allow",
   "Action": [
    "glue:GetColumnStatisticsForPartition",
    "glue:GetColumnStatisticsForTable",
     "glue:ListJobs",
     "glue:CreateSession",
     "glue:RunStatement",
     "glue:BatchCreatePartition",
     "glue:CreatePartitionIndex",
     "glue:CreateTable",
     "glue:BatchGetWorkflows",
     "glue:BatchUpdatePartition",
     "glue:BatchDeletePartition",
     "glue:GetPartition",
     "glue:GetPartitions",
     "glue:UpdateTable",
     "glue:DeleteTableVersion",
     "glue:DeleteTable",
     "glue:DeleteColumnStatisticsForPartition",
     "glue:DeleteColumnStatisticsForTable",
     "glue:DeletePartitionIndex",
     "glue:UpdateColumnStatisticsForPartition",
     "glue:UpdateColumnStatisticsForTable",
     "glue:BatchDeleteTableVersion",
     "glue:BatchDeleteTable",
     "glue:CreatePartition",
     "glue:DeletePartition",
     "glue:UpdatePartition",
     "glue:CreateBlueprint",
     "glue:CreateJob",
     "glue:CreateConnection",
     "glue:CreateCrawler",
     "glue:CreateDataQualityRuleset",
     "glue:CreateWorkflow",
     "glue:GetDatabases",
     "glue:GetTables",
     "glue:GetTable",
     "glue:SearchTables",
     "glue:NotifyEvent",
     "glue:ListSchemas",
     "glue:BatchGetJobs",
     "glue:GetConnection",
     "glue:GetDatabase"
    ],
    "Resource": [
     "*"
    ]
   },
   {
    "Sid": "AllowGlueActionsWithEnvironmentTag",
    "Effect": "Allow",
    "Action": [
     "glue:SearchTables",
     "glue:NotifyEvent",
     "glue:StartBlueprintRun",
     "glue:PutWorkflowRunProperties",
     "glue:StopCrawler",
     "glue:DeleteJob",
     "glue:DeleteWorkflow",
     "glue:UpdateCrawler",
     "glue:DeleteBlueprint",
     "glue:UpdateWorkflow",
     "glue:StartCrawler",
     "glue:ResetJobBookmark",
     "glue:UpdateJob",
     "glue:StartWorkflowRun",
     "glue:StopCrawlerSchedule",
     "glue:ResumeWorkflowRun",
     "glue:ListSchemas",
     "glue:DeleteCrawler",
     "glue:UpdateBlueprint",
     "glue:BatchStopJobRun",
     "glue:StopWorkflowRun",
     "glue:BatchGetJobs",
     "glue:BatchGetWorkflows",
     "glue:UpdateCrawlerSchedule",
     "glue:DeleteConnection",
     "glue:UpdateConnection",
     "glue:GetConnection",
     "glue:GetDatabase",
     "glue:GetTable",
     "glue:GetPartition",
     "glue:GetPartitions",
     "glue:BatchDeleteConnection",
     "glue:StartCrawlerSchedule",
     "glue:StartJobRun",
     "glue:CreateWorkflow",
     "glue:*DataQuality*"
    ],
    "Resource": "*",
    "Condition": {
     "Null": {
```

```
     "aws:ResourceTag/AmazonDataZoneEnvironment": "false"
    }
   }
  },
  {
   "Sid": "AllowGlueDefaultAccess",
   "Effect": "Allow",
   "Action": [
    "glue:BatchGet*",
    "glue:Get*",
    "glue:SearchTables",
    "glue:List*",
    "glue:RunStatement"
   ],
   "Resource": [
    "arn:aws:glue:*:*:catalog",
    "arn:aws:glue:*:*:database/default",
    "arn:aws:glue:*:*:connection/dz-sm-*",
    "arn:aws:glue:*:*:session/*"
   ]
  },
  {
   "Sid": "AllowRedshiftClusterActions",
   "Effect": "Allow",
   "Action": [
    "redshift:GetClusterCredentialsWithIAM",
    "redshift:DescribeClusters"
   ],
   "Resource": [
    "arn:aws:redshift:*:*:cluster:*",
    "arn:aws:redshift:*:*:dbname:*"
   ]
  },
  {
   "Sid": "AllowCreateClusterUser",
   "Effect": "Allow",
   "Action": [
    "redshift:CreateClusterUser"
   ],
   "Resource": [
    "arn:aws:redshift:*:*:dbuser:*"
   ]
  },
  {
   "Sid": "AllowCreateSecretActions",
   "Effect": "Allow",
   "Action": [
    "secretsmanager:CreateSecret",
    "secretsmanager:TagResource"
   ],
   "Resource": "arn:aws:secretsmanager:*:*:secret:AmazonDataZone-*",
   "Condition": {
    "StringLike": {
     "aws:ResourceTag/AmazonDataZoneDomain": "dzd_*",
     "aws:RequestTag/AmazonDataZoneDomain": "dzd_*"
    },
    "Null": {
     "aws:TagKeys": "false",
     "aws:ResourceTag/AmazonDataZoneProject": "false",
     "aws:ResourceTag/AmazonDataZoneDomain": "false",
     "aws:RequestTag/AmazonDataZoneDomain": "false",
     "aws:RequestTag/AmazonDataZoneProject": "false"
    },
    "ForAllValues:StringEquals": {
     "aws:TagKeys": [
      "AmazonDataZoneDomain",
      "AmazonDataZoneProject"
     ]
    }
   }
  },
  {
   "Sid": "ForecastOperations",
   "Effect": "Allow",
   "Action": [
    "forecast:CreateExplainabilityExport",
    "forecast:CreateExplainability",
    "forecast:CreateForecastEndpoint",
    "forecast:CreateAutoPredictor",
    "forecast:CreateDatasetImportJob",
    "forecast:CreateDatasetGroup",
    "forecast:CreateDataset",
    "forecast:CreateForecast",
    "forecast:CreateForecastExportJob",
    "forecast:CreatePredictorBacktestExportJob",
    "forecast:CreatePredictor",
    "forecast:DescribeExplainabilityExport",
    "forecast:DescribeExplainability",
    "forecast:DescribeAutoPredictor",
    "forecast:DescribeForecastEndpoint",
    "forecast:DescribeDatasetImportJob",
    "forecast:DescribeDataset",
    "forecast:DescribeForecast",
    "forecast:DescribeForecastExportJob",
    "forecast:DescribePredictorBacktestExportJob",
    "forecast:GetAccuracyMetrics",
    "forecast:InvokeForecastEndpoint",
    "forecast:GetRecentForecastContext",
    "forecast:DescribePredictor",
    "forecast:TagResource",
    "forecast:DeleteResourceTree"
   ],
   "Resource": [
    "arn:aws:forecast:*:*:*Canvas*"
   ]
  },
  {
   "Sid": "RDSOperation",
   "Effect": "Allow",
   "Action": "rds:DescribeDBInstances",
   "Resource": "*"
  },
  {
   "Sid": "AllowEventBridgeRule",
   "Effect": "Allow",
   "Action": [
    "events:PutRule"
   ],
   "Resource": "arn:aws:events:*:*:rule/*",
   "Condition": {
    "StringEquals": {
     "aws:RequestTag/sagemaker:is-canvas-data-prep-job": "true"
    }
   }
  },
  {
   "Sid": "EventBridgeOperations",
   "Effect": "Allow",
   "Action": [
    "events:DescribeRule",
    "events:PutTargets"
   ],
   "Resource": "arn:aws:events:*:*:rule/*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceTag/sagemaker:is-canvas-data-prep-job": "true"
    }
   }
  },
  {
   "Sid": "EventBridgeTagBasedOperations",
   "Effect": "Allow",
   "Action": [
    "events:TagResource"
   ],
   "Resource": "arn:aws:events:*:*:rule/*",
   "Condition": {
    "StringEquals": {
     "aws:RequestTag/sagemaker:is-canvas-data-prep-job": "true",
     "aws:ResourceTag/sagemaker:is-canvas-data-prep-job": "true"
    }
   }
  },
  {
   "Sid": "EventBridgeListTagOperation",
   "Effect": "Allow",
   "Action": "events:ListTagsForResource",
   "Resource": "*"
  },
  {
   "Sid": "AllowEMR",
   "Effect": "Allow",
   "Action": [
    "elasticmapreduce:DescribeCluster",
    "elasticmapreduce:ListInstanceGroups",
    "elasticmapreduce:ListClusters"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowSSOAction",
   "Effect": "Allow",
   "Action": [
    "sso:CreateApplicationAssignment",
    "sso:AssociateProfile"
   ],
   "Resource": "*"
  },
  {
   "Sid": "DenyNotAction",
   "Effect": "Deny",
   "NotAction": [
    "sagemaker:*",
    "sagemaker-geospatial:*",
    "sqlworkbench:*",
    "datazone:*",
    "forecast:*",
    "application-autoscaling:DeleteScalingPolicy",
    "application-autoscaling:DeleteScheduledAction",
    "application-autoscaling:DeregisterScalableTarget",
    "application-autoscaling:DescribeScalableTargets",
    "application-autoscaling:DescribeScalingActivities",
    "application-autoscaling:DescribeScalingPolicies",
    "application-autoscaling:DescribeScheduledActions",
    "application-autoscaling:PutScalingPolicy",
    "application-autoscaling:PutScheduledAction",
    "application-autoscaling:RegisterScalableTarget",
    "athena:BatchGetNamedQuery",
    "athena:BatchGetPreparedStatement",
    "athena:BatchGetQueryExecution",
    "athena:CreateNamedQuery",
    "athena:CreateNotebook",
    "athena:CreatePreparedStatement",
    "athena:CreatePresignedNotebookUrl",
    "athena:DeleteNamedQuery",
    "athena:DeleteNotebook",
    "athena:DeletePreparedStatement",
    "athena:ExportNotebook",
    "athena:GetDatabase",
    "athena:GetDataCatalog",
    "athena:GetNamedQuery",
    "athena:GetPreparedStatement",
    "athena:GetQueryExecution",
    "athena:GetQueryResults",
    "athena:GetQueryResultsStream",
    "athena:GetQueryRuntimeStatistics",
    "athena:GetTableMetadata",
    "athena:GetWorkGroup",
    "athena:ImportNotebook",
    "athena:ListDatabases",
    "athena:ListDataCatalogs",
    "athena:ListEngineVersions",
    "athena:ListNamedQueries",
    "athena:ListPreparedStatements",
    "athena:ListQueryExecutions",
    "athena:ListTableMetadata",
    "athena:ListTagsForResource",
    "athena:ListWorkGroups",
    "athena:StartCalculationExecution",
    "athena:StartQueryExecution",
    "athena:StartSession",
    "athena:StopCalculationExecution",
    "athena:StopQueryExecution",
    "athena:TerminateSession",
    "athena:UpdateNamedQuery",
    "athena:UpdateNotebook",
    "athena:UpdateNotebookMetadata",
    "athena:UpdatePreparedStatement",
    "aws-marketplace:ViewSubscriptions",
    "cloudformation:GetTemplateSummary",
    "cloudformation:ListStackResources",
    "cloudwatch:DeleteAlarms",
    "cloudwatch:DescribeAlarms",
    "cloudwatch:GetMetricData",
    "cloudwatch:GetMetricStatistics",
    "cloudwatch:ListMetrics",
    "cloudwatch:PutMetricAlarm",
    "cloudwatch:PutMetricData",
    "codebuild:BatchGetBuilds",
    "codebuild:StartBuild",
    "codecommit:BatchGetRepositories",
    "codecommit:CreateRepository",
    "codecommit:GetRepository",
    "codecommit:List*",
    "codecommit:GitPull",
    "codecommit:GitPush",
    "ec2:CreateNetworkInterface",
    "ec2:CreateNetworkInterfacePermission",
    "ec2:DeleteNetworkInterface",
    "ec2:DeleteNetworkInterfacePermission",
    "ec2:DescribeDhcpOptions",
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribeRouteTables",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSubnets",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeVpcEndpointServices",
    "ec2:DescribeVpcs",
    "ecr:BatchCheckLayerAvailability",
    "ecr:BatchGetImage",
    "ecr:CreateRepository",
    "ecr:Describe*",
    "ecr:GetAuthorizationToken",
    "ecr:GetDownloadUrlForLayer",
    "ecr:SetRepositoryPolicy",
    "ecr:CompleteLayerUpload",
    "ecr:BatchDeleteImage",
    "ecr:UploadLayerPart",
    "ecr:DeleteRepositoryPolicy",
    "ecr:InitiateLayerUpload",
    "ecr:DeleteRepository",
    "ecr:PutImage",
    "ecr:StartImageScan",
    "ecr:TagResource",
    "ecr:UntagResource",
    "elastic-inference:Connect",
    "elasticfilesystem:DescribeFileSystems",
    "elasticfilesystem:DescribeMountTargets",
    "elasticmapreduce:DescribeCluster",
    "elasticmapreduce:ListInstanceGroups",
    "elasticmapreduce:ListClusters",
    "events:PutRule",
    "events:DescribeRule",
    "events:PutTargets",
    "events:TagResource",
    "events:ListTagsForResource",
    "fsx:DescribeFileSystems",
    "glue:SearchTables",
    "glue:NotifyEvent",
    "glue:StartBlueprintRun",
    "glue:PutWorkflowRunProperties",
    "glue:StopCrawler",
    "glue:DeleteJob",
    "glue:DeleteWorkflow",
    "glue:UpdateCrawler",
    "glue:DeleteBlueprint",
    "glue:UpdateWorkflow",
    "glue:StartCrawler",
    "glue:ResetJobBookmark",
    "glue:UpdateJob",
    "glue:StartWorkflowRun",
    "glue:StopCrawlerSchedule",
    "glue:ResumeWorkflowRun",
    "glue:DeleteCrawler",
    "glue:UpdateBlueprint",
    "glue:BatchStopJobRun",
    "glue:StopWorkflowRun",
    "glue:BatchGet*",
    "glue:UpdateCrawlerSchedule",
    "glue:DeleteConnection",
    "glue:UpdateConnection",
    "glue:Get*",
    "glue:BatchDeleteConnection",
    "glue:StartCrawlerSchedule",
    "glue:StartJobRun",
    "glue:CreateWorkflow",
    "glue:*DataQuality*",
    "glue:List*",
    "glue:CreateSession",
    "glue:RunStatement",
    "glue:BatchCreatePartition",
    "glue:CreateDatabase",
    "glue:CreatePartitionIndex",
    "glue:CreateTable",
    "glue:BatchUpdatePartition",
    "glue:BatchDeletePartition",
    "glue:UpdateTable",
    "glue:DeleteTableVersion",
    "glue:DeleteTable",
    "glue:DeleteColumnStatisticsForPartition",
    "glue:DeleteColumnStatisticsForTable",
    "glue:DeletePartitionIndex",
    "glue:UpdateColumnStatisticsForPartition",
    "glue:UpdateColumnStatisticsForTable",
    "glue:BatchDeleteTableVersion",
    "glue:BatchDeleteTable",
    "glue:CreatePartition",
    "glue:DeletePartition",
    "glue:UpdatePartition",
    "glue:CreateBlueprint",
    "glue:CreateJob",
    "glue:CreateConnection",
    "glue:CreateCrawler",
    "groundtruthlabeling:*",
    "iam:CreateServiceLinkedRole",
    "iam:GetRole",
    "iam:ListRoles",
    "iam:PassRole",
    "kms:DescribeKey",
    "kms:ListAliases",
    "kms:Decrypt",
    "kms:ListKeys",
    "kms:Encrypt",
    "kms:GenerateDataKey",
    "kms:RetireGrant",
    "lakeformation:GetDataAccess",
    "lambda:ListFunctions",
    "lambda:InvokeFunction",
    "logs:CreateLogDelivery",
    "logs:CreateLogGroup",
    "logs:CreateLogStream",
    "logs:DeleteLogDelivery",
    "logs:Describe*",
    "logs:GetLogDelivery",
    "logs:GetLogEvents",
    "logs:ListLogDeliveries",
    "logs:PutLogEvents",
    "logs:UpdateLogDelivery",
    "ram:AcceptResourceShareInvitation",
    "rds:DescribeDBInstances",
    "redshift:CreateClusterUser",
    "redshift:GetClusterCredentials",
    "redshift:GetClusterCredentialsWithIAM",
    "redshift:DescribeClusters",
    "redshift-data:BatchExecuteStatement",
    "redshift-data:CancelStatement",
    "redshift-data:DescribeStatement",
    "redshift-data:DescribeTable",
    "redshift-data:ExecuteStatement",
    "redshift-data:GetStatementResult",
    "redshift-data:ListSchemas",
    "redshift-data:ListTables",
    "redshift-serverless:ListNamespaces",
    "redshift-serverless:ListWorkgroups",
    "redshift-serverless:GetNamespace",
    "redshift-serverless:GetWorkgroup",
    "redshift-serverless:GetCredentials",
    "s3:GetBucketAcl",
    "s3:PutObjectAcl",
    "s3:GetObject",
    "s3:PutObject",
    "s3:DeleteObject",
    "s3:AbortMultipartUpload",
    "s3:CreateBucket",
    "s3:GetBucketLocation",
    "s3:ListBucket",
    "s3:ListAllMyBuckets",
    "s3:GetBucketCors",
    "s3:PutBucketCors",
    "s3:DeleteObjectVersion",
    "s3:PutObjectRetention",
    "s3:ReplicateObject",
    "s3:RestoreObject",
    "secretsmanager:ListSecrets",
    "secretsmanager:DescribeSecret",
    "secretsmanager:GetSecretValue",
    "secretsmanager:CreateSecret",
    "secretsmanager:PutResourcePolicy",
    "secretsmanager:TagResource",
    "servicecatalog:Describe*",
    "servicecatalog:List*",
    "servicecatalog:ScanProvisionedProducts",
    "servicecatalog:SearchProducts",
    "servicecatalog:SearchProvisionedProducts",
    "servicecatalog:ProvisionProduct",
    "servicecatalog:TerminateProvisionedProduct",
    "servicecatalog:UpdateProvisionedProduct",
    "sns:ListTopics",
    "sns:Subscribe",
    "sns:CreateTopic",
    "sns:Publish",
    "states:DescribeExecution",
    "states:GetExecutionHistory",
    "states:StartExecution",
    "states:StopExecution",
    "states:UpdateStateMachine",
    "tag:GetResources",
    "sso:CreateApplicationAssignment",
    "sso:AssociateProfile"
   ],
   "Resource": "*"
  }
 ]
}
```

Amazon DataZone updates to AWS managed policies

View details about updates to AWS managed policies for Amazon DataZone since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the Amazon DataZone Document history page.

Change	Description	Date
AmazonDataZoneExecutionRolePolicy and AmazonDataZoneFullUserAccess - policy update	Policy update to the **AmazonDataZoneExecutionRolePolicy** and **AmazonDataZoneFullUserAccess** to enable support for the data lineage and fine grained access control APIs.	June 27th, 2024
AmazonDataZoneGlueManageAccessRolePolicy - policy update	Policy update to the **AmazonDataZoneGlueManageAccessRolePolicy** that adds IAM permissions required for the self-subscribe functionality in Amazon DataZone in order to scope down the permissions granting in lake formation. With the self-subscribe functionality, the lake formation permissions can only be granted to tagged resourcese.	June 14th, 2024

Change	Description	Date
AmazonDataZoneDomainExecutionRolePolicy - policy update	Policy update to the **AmazonDataZoneDomainExecutionRolePolicy** that adds new APIs to Amazon DataZone that enable users to configure actions for their Amazon DataZone environments.	June 14th, 2024
AmazonDataZoneFullAccess - policy update	Policy update to the **AmazonDataZoneFullAccess** that enables the Amazon DataZone management console to create secrets on user's behalf with both domain and project tags. Also including the `ram:ListResourceSharePermissions` action to enable administrations from the domain owner account to view the account association status of the associated accounts.	June 14th, 2024

Change	Description	Date
AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary - new permissions boundary	New permissions boundary called **AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary** . When you create an Amazon SageMaker environment via the Amazon DataZone data portal, Amazon DataZone applies this permissions boundary to the IAM roles that are produced during environment creation. The permissions boundary limits the scope of the roles that Amazon DataZone creates and any roles that you add.	April 30th, 2024
AmazonDataZoneSageMakerAccess - new policy	New policy called **AmazonDataZoneSageMakerAccess** gives Amazon DataZone permissions to publish Amazon SageMaker assets to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to the Amazon SageMaker published assets in the catalog.	April 30th, 2024

Change	Description	Date
AmazonDataZoneFullAccess - policy update	An update to the **AmazonDataZoneFullAccess** policy that adds access to `DescribeSecurityGroups` action to improve the usability for account administrators configuring blueprints in the console and `GetPolicy` action to help retrieve information about the specified managed policy.	April 30th, 2024
AmazonDataZoneSageMakerProvisioning - new policy	New policy called **AmazonDataZoneSageMakerProvisioning** grants Amazon DataZone the permissions required to interoperate with Amazon SageMaker.	April 30th, 2024
AmazonDataZoneS3Manage-<region>-<domainId> - new role	New role called **AmazonDataZoneS3Manage-<region>-<domainId>** that is used when Amazon DataZone calls AWS Lake Formation to register an Amazon Simple Storage Service (Amazon S3) location. AWS Lake Formation assumes this role when accessing the data in that location.	April 1st, 2024

Change	Description	Date
AmazonDataZoneGlueManageAccessRolePolicy - Policy update	Updated the **AmazonDataZoneGlueManageAccessRolePolicy** to enable support for permissions that allow Amazon DataZone to enable publishing and access grants to data.	April 1st, 2024
AmazonDataZoneDomainExecutionRolePolicy and AmazonDataZoneFullUserAccess - Policy update	Updated the **AmazonDataZoneDomainExecutionRolePolicy** and **AmazonDataZoneFullUserAccess** to enable support for the `CancelMetadataGenerationRun` API.	March 29, 2024
AmazonDataZoneFullAccess - Policy update	Updated the `AmazonDataZoneFullAccess` to enable users to choose their secrets, clusters, vpc's, and subnets in the Amazon DataZone management console rather than type them in a text box.	March 13, 2024

Change	Description	Date
AmazonDataZoneDomainExecutionRolePolicy - Policy update	Updated the **AmazonDataZoneDomainExecutionRolePolicy** to enable support for the `ListEnvironmentBlueprintConfigurationSummaries` API that is required for creating environment profiles by identifying which blueprints are enabled in which account and region.	February 01, 2024
AmazonDataZoneGlueManageAccessRolePolicy - Policy update	Updated the **AmazonDataZoneGlueManageAccessRolePolicy** to enable support for the AWS Lake Formation hybrid mode.	December 14, 2023
AmazonDataZoneFullUserAccess and AmazonDataZoneDomainExecutionRolePolicy - Policy updates	Updated the **AmazonDataZoneFullUserAccess** and the **AmazonDataZoneDomainExecutionRolePolicy** policies to support the generative AI-powered data descriptions functionality in Amazon DataZone.	November 28, 2023

Change	Description	Date
AmazonDataZoneEnvironmentRolePermissionsBoundary - Policy update	Amazon DataZone made an update to the **AmazonDataZoneEnvironmentRolePermissionsBoundary** managed policy that consists of an additional `athena:GetQueryResultsStream` permission scoped down with the `ResourceTag` condition .	November 17, 2023
AmazonDataZoneRedshiftManageAccessRolePolicy - Policy update	Amazon DataZone updated the **AmazonDataZoneRedshiftManageAccessRolePolicy** by removing the check on organization ID for the `redshift:AssociateDataShareConsumer` action. This enables you to share resource across AWS organizations.	November 16, 2023
AmazonDataZoneFullUserAccess - Policy update	Amazon DataZone updated the **AmazonDataZoneFullUserAccess** policy that grants full access to Amazon DataZone, but it does not allow the management of domains, users, or associated accounts.	October 02, 2023
AmazonDataZonePortalFullAccessPolicy - policy deprecated	Amazon DataZone deprecated the **AmazonDataZonePortalFullAccessPolicy**.	September 29, 2023

Change	Description	Date
AmazonDataZonePreviewConsoleFullAccess - policy deprecated	Amazon DataZone deprecated the **AmazonDataZonePreviewConsoleFullAccess**.	September 29, 2023
AmazonDataZoneDomainExecutionRolePolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneDomainExecutionRolePolicy**. This is the default policy for the Amazon DataZone `AmazonDataZoneDomainExecutionRole` service role. This role is used by Amazon DataZone to catalog, discover, govern, share, and analyze data in the Amazon DataZone domain. You can attach the `AmazonDataZoneDomainExecutionRolePolicy` policy to your `AmazonDataZoneDomainExecutionRole` .	September 25, 2023
AmazonDataZoneCrossAccountAdmin - New policy	Amazon DataZone added a new policy called **AmazonDataZoneCrossAccountAdmin** that enables users to work with Amazon DataZone and its associated accounts.	September 19, 2023

Change	Description	Date
AmazonDataZoneFullUserAccess - New policy	Amazon DataZone added a new policy called **AmazonDataZoneFullUserAccess** that grants full access to Amazon DataZone, but it does not allow the management of domains, users, or associated accounts.	September 12, 2023
AmazonDataZoneRedshiftManageAccessRolePolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneRedshiftManageAccessRolePolicy** that grants permissions to allow Amazon DataZone to enable publishing and access grants to data.	September 12, 2023
AmazonDataZoneGlueManageAccessRolePolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneGlueManageAccessRolePolicy** that grants Amazon DataZone permissions to publish AWS Glue data to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to AWS Glue published assets in the catalog.	September 12, 2023

Change	Description	Date
AmazonDataZoneRedshiftGlueProvisioningPolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneRedshiftGlueProvisioningPolicy** that grants Amazon DataZone the permissions required to interoperate with the supported data sources.	September 12, 2023
AmazonDataZoneEnvironmentRolePermissionsBoundary - New policy	Amazon DataZone added a new policy called **AmazonDataZoneEnvironmentRolePermissionsBoundary** that limits the provisioned IAM principal to which it is attached.	September 12, 2023
AmazonDataZoneFullAccess - New policy	Amazon DataZone added a new policy called **AmazonDataZoneFullAccess** that provides full access to Amazon DataZone via the AWS Management Console.	September 12, 2023
Managed policy update	Updates to the **AmazonDataZonePreviewConsoleFullAccess** managed policy that consists of an additional `iam:GetPolicy` permissions.	June 13, 2023
Amazon DataZone started tracking changes	Amazon DataZone started tracking changes for its AWS managed policies.	March 20, 2023

IAM roles for Amazon DataZone

Topics

- AmazonDataZoneProvisioningRole-<domainAccountId>
- AmazonDataZoneDomainExecutionRole
- AmazonDataZoneGlueAccess-<region>-<domainId>
- AmazonDataZoneRedshiftAccess-<region>-<domainId>
- AmazonDataZoneS3Manage-<region>-<domainId>
- AmazonDataZoneSageMakerManageAccessRole-<region>-<domainId>
- AmazonDataZoneSageMakerProvisioningRole-<domainAccountId>

AmazonDataZoneProvisioningRole-<domainAccountId>

The `AmazonDataZoneProvisioningRole-<domainAccountId>` has the `AmazonDataZoneRedshiftGlueProvisioningPolicy` attached. This role grants Amazon DataZone the permissions required to interoperate with AWS Glue and Amazon Redshift.

The default `AmazonDataZoneProvisioningRole-<domainAccountId>` has the following trust policy attached:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "datazone.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "{{domain_account}}"
        }
      }
    }
  ]
}
```

AmazonDataZoneDomainExecutionRole

The **AmazonDataZoneDomainExecutionRole** has the AWS managed policy **AmazonDataZoneDomainExecutionRolePolicy** attached. Amazon DataZone creates this role for you on your behalf. For certain actions in the data portal, Amazon DataZone assumes this role in the account in which the role is created and checks that this role is authorized to perform the action.

The **AmazonDataZoneDomainExecutionRole** role is required in the AWS account that hosts your Amazon DataZone domain. This role is automatically created for you when you create your Amazon DataZone domain.

The default **AmazonDataZoneDomainExecutionRole** role has the following trust policy.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "datazone.amazonaws.com"
            },
            "Action": [
                "sts:AssumeRole",
                "sts:TagSession"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "{{source_account_id}}"
                },
                "ForAllValues:StringLike": {
                    "aws:TagKeys": [
                        "datazone*"
                    ]
                }
            }
        }
    ]
}
```

AmazonDataZoneGlueAccess-<region>-<domainId>

The `AmazonDataZoneGlueAccess-<region>-<domainId>` role has the `AmazonDataZoneGlueManageAccessRolePolicy` attached. This role grants Amazon DataZone permissions to publish AWS Glue data to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to AWS Glue published assets in the catalog.

The default `AmazonDataZoneGlueAccess-<region>-<domainId>` role has the following trust policy attached:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "datazone.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "{{domain_account}}"
            },
            "ArnEquals": {
                "aws:SourceArn": "arn:aws:datazone:{{region}}:
{{domain_account}}:domain/{{root_domain_id}}"
            }
        }
    }
  ]
}
```

AmazonDataZoneRedshiftAccess-<region>-<domainId>

The `AmazonDataZoneRedshiftAccess-<region>-<domainId>` role has the `AmazonDataZoneRedshiftManageAccessRolePolicy` attached. This role grants Amazon DataZone permissions to publish Amazon Redshift data to the catalog. It also gives Amazon

DataZone permissions to grant access or revoke access to Amazon Redshift or Amazon Redshift Serverless published assets in the catalog.

The default `AmazonDataZoneRedshiftAccess-<region>-<domainId>` role has the following inline permissions policy attached:

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid": "RedshiftSecretStatement",
         "Effect":"Allow",
         "Action":"secretsmanager:GetSecretValue",
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "secretsmanager:ResourceTag/AmazonDataZoneDomain":"{{domainId}}"
            }
         }
      }
   ]
}
```

The default `AmazonDataZoneRedshiftManageAccessRole<timestamp>` has the following trust policy attached:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "datazone.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "{{domain_account}}"
            },
            "ArnEquals": {
                "aws:SourceArn": "arn:aws:datazone:{{region}}:
{{domain_account}}:domain/{{root_domain_id}}"
            }
        }
    }
  ]
}
```

AmazonDataZoneS3Manage-<region>-<domainId>

The AmazonDataZoneS3Manage-<region>-<domainId> is used when Amazon DataZone calls AWS Lake Formation to register an Amazon Simple Storage Service (Amazon S3) location. AWS Lake Formation assumes this role when accessing the data in that location. For more information, see Requirements for roles used to register locations.

This role has the following inline permissions policy attached.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "LakeFormationDataAccessPermissionsForS3",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "{{accountId}}"
                }
            }
        },
        {
            "Sid": "LakeFormationDataAccessPermissionsForS3ListBucket",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "{{accountId}}"
                }
            }
        },
        {
            "Sid": "LakeFormationDataAccessPermissionsForS3ListAllMyBuckets",
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Resource": "arn:aws:s3:::*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "{{accountId}}"
                }
            }
        },
        {
            "Sid": "LakeFormationExplicitDenyPermissionsForS3",
            "Effect": "Deny",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::[[BucketNames]]/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "{{accountId}}"
                }
            }
        },
        {
            "Sid": "LakeFormationExplicitDenyPermissionsForS3ListBucket",
            "Effect": "Deny",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::[[BucketNames]]"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount": "{{accountId}}"
                }
            }
        }
    ]
}
```

The AmazonDataZoneS3Manage-<region>-<domainId> has the following trust policy attached:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "TrustLakeFormationForDataLocationRegistration",
            "Effect": "Allow",
            "Principal": {
                "Service": "lakeformation.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "{{source_account_id}}"
                }
            }
        }
    ]
}
```

AmazonDataZoneSageMakerManageAccessRole-<region>-<domainId>

The `AmazonDataZoneSageMakerManageAccessRole` role has the `AmazonDataZoneSageMakerAccess`, the `AmazonDataZoneRedshiftManageAccessRolePolicy`, and the `AmazonDataZoneGlueManageAccessRolePolicy` attached. This role grants Amazon DataZone permissions to publish and manage subscriptions for data lake, data warehouse, and Amazon Sagemaker assets.

The `AmazonDataZoneSageMakerManageAccessRole` role has the following inline policy attached:

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid": "RedshiftSecretStatement",
         "Effect":"Allow",
         "Action":"secretsmanager:GetSecretValue",
         "Resource":"*",
         "Condition":{
            "StringEquals":{
               "secretsmanager:ResourceTag/AmazonDataZoneDomain":"{{domainId}}"
            }
         }
      }
   ]
}
```

The `AmazonDataZoneSageMakerManageAccessRole` role has the following trust policy attached:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DatazoneTrustPolicyStatement",
      "Effect": "Allow",
      "Principal": {
        "Service":  ["datazone.amazonaws.com",
                   "sagemaker.amazonaws.com"]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "{{domain_account}}"
            },
            "ArnEquals": {
                "aws:SourceArn": "arn:aws:datazone:{{region}}:
{{domain_account}}:domain/{{root_domain_id}}"
            }
        }
    }
]
}
```

AmazonDataZoneSageMakerProvisioningRole-<domainAccountId>

The `AmazonDataZoneSageMakerProvisioningRole` role has the `AmazonDataZoneSageMakerProvisioning` and the `AmazonDataZoneRedshiftGlueProvisioningPolicy` attached. This role grants Amazon DataZone permissions required to interoperate with AWS Glue, Amazon Redshift, and Amazon Sagemaker.

The `AmazonDataZoneSageMakerProvisioningRole` role has the following inline policy attached:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "SageMakerStudioTagOnCreate",
            "Effect": "Allow",
            "Action": [
                "sagemaker:AddTags"
            ],
            "Resource": "arn:aws:sagemaker:*:{{AccountId}}:*/*",
            "Condition": {
                "Null": {
                    "sagemaker:TaggingAction": "false"
                }
            }
        }
    ]
}
```

The `AmazonDataZoneSageMakerProvisioningRole` role has the following trust policy attached:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DataZoneTrustPolicyStatement",
      "Effect": "Allow",
      "Principal": {
        "Service": "datazone.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "{{domain_account}}"
        }
      }
    }
  ]
}
```

Temporary Credentials

Some AWS services don't work when you sign in using temporary credentials. For additional information, including which AWS services work with temporary credentials, see AWS services that work with IAM in the *IAM User Guide*.

You are using temporary credentials if you sign in to the AWS Management Console using any method except a user name and password. For example, when you access AWS using your company's single sign-on (SSO) link, that process automatically creates temporary credentials. You also automatically create temporary credentials when you sign in to the console as a user and then switch roles. For more information about switching roles, see Switching to a role (console) in the *IAM User Guide*.

You can manually create temporary credentials using the AWS CLI or AWS API. You can then use those temporary credentials to access AWS. AWS recommends that you dynamically generate temporary credentials instead of using long-term access keys. For more information, see Temporary security credentials in IAM.

Principal permissions

When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see Actions, Resources, and Condition Keys for AWS Documentation Essentials in the *Service Authorization Reference*.

Compliance validation for Amazon DataZone

To learn whether an AWS service is within the scope of specific compliance programs, see AWS services in Scope by Compliance Program and choose the compliance program that you are interested in. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading Reports in AWS Artifact.

Your compliance responsibility when using AWS services is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying baseline environments on AWS that are security and compliance focused.
- Architecting for HIPAA Security and Compliance on Amazon Web Services – This whitepaper describes how companies can use AWS to create HIPAA-eligible applications.

> **Note**
>
> Not all AWS services are HIPAA eligible. For more information, see the HIPAA Eligible Services Reference.

- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Customer Compliance Guides – Understand the shared responsibility model through the lens of compliance. The guides summarize the best practices for securing AWS services and map

the guidance to security controls across multiple frameworks (including National Institute of Standards and Technology (NIST), Payment Card Industry Security Standards Council (PCI), and International Organization for Standardization (ISO)).

- Evaluating Resources with Rules in the *AWS Config Developer Guide* – The AWS Config service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS. Security Hub uses security controls to evaluate your AWS resources and to check your compliance against security industry standards and best practices. For a list of supported services and controls, see Security Hub controls reference.
- Amazon GuardDuty – This AWS service detects potential threats to your AWS accounts, workloads, containers, and data by monitoring your environment for suspicious and malicious activities. GuardDuty can help you address various compliance requirements, like PCI DSS, by meeting intrusion detection requirements mandated by certain compliance frameworks.
- AWS Audit Manager – This AWS service helps you continuously audit your AWS usage to simplify how you manage risk and compliance with regulations and industry standards.

Security Best Practices for Amazon DataZone

Amazon DataZone provides a number of security features to consider as you develop and implement your own security policies. The following best practices are general guidelines and don't represent a complete security solution. Because these best practices might not be appropriate or sufficient for your environment, treat them as helpful considerations rather than prescriptions.

Implement least privilege access

When granting permissions, you decide who is getting what permissions to which Amazon DataZone resources. You enable specific actions that you want to allow on those resources. Therefore you should grant only the permissions that are required to perform a task. Implementing least privilege access is fundamental in reducing security risk and the impact that could result from errors or malicious intent.

Use IAM roles

Producer and client applications must have valid credentials to access Amazon DataZone resources. You should not store AWS credentials directly in a client application or in an Amazon S3 bucket.

These are long-term credentials that are not automatically rotated and could have a significant business impact if they are compromised.

Instead, you should use an IAM role to manage temporary credentials for your producer and client applications to access Amazon DataZone resources. When you use a role, you don't have to use long-term credentials (such as a user name and password or access keys) to access other resources.

For more information, see the following topics in the *IAM User Guide*:

- IAM Roles
- Common Scenarios for Roles: Users, Applications, and Services

Implement Server-Side Encryption in Dependent Resources

Data at rest and data in transit can be encrypted in Amazon DataZone.

Use CloudTrail to Monitor API Calls

Amazon DataZone is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon DataZone.

Using the information collected by CloudTrail, you can determine the request that was made to Amazon DataZone, the IP address from which the request was made, who made the request, when it was made, and additional details.

Resilience in Amazon DataZone

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

In addition to the AWS global infrastructure, Amazon DataZone offers several features to help support your data resiliency and backup needs.

Topics

Data source resilience

During an Amazon DataZone availability event, `DataSource` jobs will periodically retry for up to 24 hours. If a job fails due to a misconfiguration, a `DataSourceRunFailed` event will be emitted. If the Amazon DataZone domain is configured with a KMS key, and the AmazonDataZoneDomainExecutionRole loses access to this key during a job run, the run will end in the `INACCESSIBLE` state. Once KMS access is restored, the job should be manually updated to trigger the transition back to a useable state.

Asset resilience

In Amazon DataZone, assets are versioned. If a version of an asset needs to be rolled back, you can create a new version using content of the last stable version. An asset version can be published. A published version of an asset cannot be edited, except by publishing a new version. A published asset (aka listing) can be subscribed to. To prevent new subscriptions to an asset, it can be unpublished. Un-publishing an asset does not have an effect on the existing subscriptions. Deleting an asset will delete all unpublished versions of the asset. Published versions of the asset must be deleted separately. A published version of an asset can be deleted only if there are no subscriptions.

Asset type and metadata form resilience

In Amazon DataZone, asset types and metadata form types are versioned. An asset type cannot be deleted if it is in use by an asset. A metadata form type cannot be deleted if it is in use by an asset type or an asset. If you don't want specific metadata-form-type to be used for curation, you can disable them which doesn't affect the ones it's already attached to.

Glossary resilience

In Amazon DataZone, glossaries and glossary terms cannot be deleted if they are in use. If you don't want specific glossary or glossary-term to be used for curation, you can disable them which doesn't affect the ones it's already attached to.

Global search resilience

In Amazon DataZone, published assets (aka listings) can be discovered through global search. Publishing of an asset can be rolled back by unpublishing the asset. Unpublishing an asset does not affect existing subscriptions. A published asset can be rolled back to a particular version of the asset by republishing that version. This will not effect existing subscriptions.

Subscription resilience

In Amazon DataZone, subscriptionGrant fulfillment will attempt two retires before failing. If it fails, it must be manually deleted to retry. If Amazon DataZone cannot revoke permissions for a subscription, deleting the subscription may fail. The underlying error should be addressed, or the `retainPermissions` flag can be used in the `DeleteSubscriptionGrant` API operation to force deletion of the grant from Amazon DataZone without revoking the permissions.

If the Amazon DataZone domain is configured with a KMS key, and the `AmazonDataZoneDomainExecutionRole` loses access to this key during the `SubscriptionGrant` workflow, the grant is marked `INACCESSIBLE`. Once KMS access is restored, the `INACCESSIBLE` grants must be deleted and recreate.

Environment resilience

If the Amazon DataZone domain is configured with a KMS key, and the `AmazonDataZoneDomainExecutionRole` loses access to this key during the environment workflow, the environment will be marked `INACCESSIBLE`. Once KMS access is restored, the `INACCESSIBLE` environment must be deleted and recreated. Environment creation will attempt two retires before failing. If it fails, it must be manually deleted to retry. If the environment workflow fails, the environment will enter a failed state. At this point, it can only be deleted and recreated.

Environment blueprint resilience

In Amazon DataZone, an environment blueprint cannot be deleted if there are any underlying environment profiles.

Project resilience

In Amazon DataZone, a project cannot be deleted if there are any contained environments.

RAM resilience

For RAM resilience information, see https://docs.aws.amazon.com/ram/latest/userguide/security-disaster-recovery-resiliency.html.

User profile management resilience

For user profile resilience information, see AWS Identity Center.

Domain resilience

In Amazon DataZone, a domain cannot be deleted if it contains projects or data sources.

Infrastructure Security in Amazon DataZone

As a managed service, Amazon DataZone is protected by AWS global network security. For information about AWS security services and how AWS protects infrastructure, see AWS Cloud Security. To design your AWS environment using the best practices for infrastructure security, see Infrastructure Protection in *Security Pillar AWS Well‐Architected Framework*.

You use AWS published API calls to access Amazon DataZone through the network. Clients must support the following:

- Transport Layer Security (TLS). We require TLS 1.2 and recommend TLS 1.3.
- Cipher suites with perfect forward secrecy (PFS) such as DHE (Ephemeral Diffie-Hellman) or ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

Cross-service confused deputy prevention in Amazon DataZone

The confused deputy problem is a security issue where an entity that doesn't have permission to perform an action can coerce a more-privileged entity to perform the action. In AWS, cross-service impersonation can result in the confused deputy problem. Cross-service impersonation can occur when one service (the calling service) calls another service (the called service). The calling service can be manipulated to use its permissions to act on another customer's resources in a way it should not otherwise have permission to access. To prevent this, AWS provides tools that help you protect your data for all services with service principals that have been given access to resources in your account.

We recommend using the aws:SourceAccount global condition context key in resource policies to limit the permissions that Amazon DataZone gives another service to the resource. Use aws:SourceAccount if you want to allow any resource in that account to be associated with the cross-service use.

Configuration and vulnerability analysis for Amazon DataZone

AWS handles basic security tasks like guest operating system (OS) and database patching, firewall configuration, and disaster recovery. These procedures have been reviewed and certified by the appropriate third parties. For more information, see the AWS shared responsibility model.

Domains to add to your allow list

For the Amazon DataZone data portal to access the Amazon DataZone service, you must add the following domains to the allow list on the network from which the data portal is trying to access the service.

- `*.api.aws`
- `*.on.aws`

Monitoring Amazon DataZone

Monitoring is an important part of maintaining the reliability, availability, and performance of Amazon DataZone and your other AWS solutions. AWS provides the following monitoring tools to watch Amazon DataZone, report when something is wrong, and take automatic actions when appropriate:

- *Amazon CloudWatch* monitors your AWS resources and and the applications you run on AWS in real time. You can collect and track metrics, create customized dashboards, and set alarms that notify you or take actions when a specified metric reaches a threshold that you specify. For example, you can have CloudWatch track CPU usage or other metrics of your Amazon EC2 instances and automatically launch new instances when needed. For more information, see the Amazon CloudWatch User Guide.
- *Amazon CloudWatch Logs* enables you to monitor, store, and access your log files from Amazon EC2 instances, CloudTrail, and other sources. CloudWatch Logs can monitor information in the log files and notify you when certain thresholds are met. You can also archive your log data in highly durable storage. For more information, see the Amazon CloudWatch Logs User Guide.
- *Amazon EventBridge* can be used to automate your AWS services and respond automatically to system events, such as application availability issues or resource changes. Events from AWS services are delivered to EventBridge in near real time. You can write simple rules to indicate which events are of interest to you and which automated actions to take when an event matches a rule. For more information, see Amazon EventBridge User Guide.
- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.

Monitoring Amazon DataZone events in Amazon EventBridge

You can monitor Amazon DataZone events in EventBridge, which delivers a stream of real-time data from your own applications, software-as-a-service (SaaS) applications, and AWS services. EventBridge routes that data to targets such as AWS Lambda and Amazon Simple Notification Service. These events are the same as those that appear in Amazon CloudWatch Events, which delivers a near real-time stream of system events that describe changes in AWS resources.

For more information, see Working with events via Amazon EventBridge default bus.

Logging Amazon DataZone API calls using AWS CloudTrail

Amazon DataZone is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon DataZone. CloudTrail captures all API calls for Amazon DataZone as events. The calls captured include calls from the Amazon DataZone console and code calls to the Amazon DataZone API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Amazon DataZone. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to Amazon DataZone, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Amazon DataZone information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in the Amazon DataZone management console, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing events with CloudTrail Event history.

For an ongoing record of events in your AWS account, including events for Amazon DataZone, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple regions and Receiving CloudTrail log files from multiple accounts

All Amazon DataZone actions are logged by CloudTrail.

Troubleshooting Amazon DataZone

If you encounter access-denied issues or similar difficulties when working with Amazon DataZone consult the topics in this section.

Troubleshooting AWS Lake Formation permissions for Amazon DataZone

This section contains troubleshooting instructions for issues that you might encounter when you Configure Lake Formation permissions for Amazon DataZone.

Error message in the Data Portal	Resolution
Unable to assume the Data Access Role.	This error is displayed when Amazon DataZone is unable to assume the **AmazonDataZoneGlueDataAccessRole** that you used to enable the **DefaultDataLakeBlueprint** in your account. To fix the issue, go to the AWS IAM console in the account where your data asset exists and make sure that the **AmazonDataZoneGlueDataAccessRole** has the right trust relationship with the Amazon DataZone service principal. For more information, see AmazonDataZoneGlueAccess-<region>-<domainId>
The Data Access Role does not have the necessary permissions to read the metadata of the asset you are trying to subscribe.	This error is displayed when Amazon DataZone successfully assumes the **AmazonDataZoneGlueDataAccessRole** role, but the role does not have the necessary permissions. To fix the issue, go to the AWS IAM console in the account where your data asset exists and make sure that the role has the **AmazonDataZoneGlueManageAccessRolePolicy** attached it. For more information, see AmazonDataZoneGlueAccess-<region>-<domainId>.

Error message in the Data Portal	Resolution
Asset is a resource link. Amazon DataZone does not support subscriptions to resource links.	This error is displayed when the asset you are trying to publish to Amazon DataZone is a resource link to an AWS Glue table.

Error message in the Data Portal	Resolution
Asset is not managed by AWS Lake Formation.	This error indicates that the AWS Lake Formation permissions are not enforced on the asset that you want to publish. This can happen in the following cases. • The Amazon S3 location of the asset is not registered in AWS Lake Formation. To fix the issue, log into your AWS Lake Formation console in the account where the table exists and register the Amazon S3 location either in AWS Lake Formation mode or Hybrid mode. For more information, see Registering an Amazon S3 location. There are several scenarios that require further modifications. These include encrypted AmazonS3 buckets or a cross-account S3 bucket and an AWS Glue Catalog setup. In such cases, modifications in KMS and/or S3 settings may be necessary. For more information, see Registering an encrypted Amazon S3 location. • The Amazon S3 location is registered in AWS Lake Formation mode but **IAMAllowedPrincipal** is added to the table's permissions. To fix the issue, you can either remove the **IAMAllowedPrincipal** from the table's permissions or register the S3 location in Hybrid mode. For more information, see About upgrading to the Lake Formation permissions model. If your S3 location is encrypted or the S3 location is in a different accout than than your AWS Glue table, follow the instructions in Registering an encrypted Amazon S3 location.

Error message in the Data Portal	Resolution
Data Access role does not have necessary Lake Formation permissions to grant access to this asset.	This error indicates that the **AmazonDataZoneGlueDataAccessRole** that you are using to enable the **DefaultDataLakeBlueprint** in your account does not have the necessary permissions for Amazon DataZone to manage permissions on the published asset. You can resolve the issue by either adding the **AmazonDataZoneGlueDataAccessRole** as the AWS Lake Formation administrator or by granting the following permissions to the **AmazonDataZoneGlueDataAccessRole** on the asset that you want to publish. • Describe and Describe grantable permissions on the database where the asset exist • Describe, Select, Describe Grantable, Select Grantable permissions on the all the assets in the database the acecss to which you wanto Amazon DataZone to manage on your behalf.

Quotas for Amazon DataZone

Your AWS account has default quotas, formerly referred to as limits, for each AWS service. Unless otherwise noted, each quota is region-specific.

Amazon DataZone has the following quotas and limits.

Resource	Description	Value
Data Asset Types	The maximum number of data asset types that can be created in a DataZone domain	1000
Data assets	The maximum number of data assets that can be created in an Amazon DataZone domain	1 million
Glossaries	The maximum number of business glossaries you can create in a domain	1000
Business glossary terms	The maximum number of total business glossary terms you can create in a domain	10000
Environments in a domain	The maximum number of environments in an Amazon DataZone domain	500

Document history for the Amazon DataZone User Guide

The following table describes the documentation releases for Amazon DataZone.

Change	Description	Date
AmazonDataZoneExecutionRolePolicy and AmazonDataZoneFullUserAccess - policy update	Policy update to the **AmazonDataZoneExecutionRolePolicy** and **AmazonDataZoneFullUserAccess** to enable support for the data lineage and fine grained access control APIs. For more information, see Amazon DataZone updates to AWS managed policies.	June 27, 2024
AmazonDataZoneGlueManageAccessRolePolicy - policy update	Policy update to the **AmazonDataZoneGlueManageAccessRolePolicy** that adds IAM permissions required for the self-subscribe functionality in Amazon DataZone in order to scope down the permissions granting in lake formation. With the self-subscribe functionality, the lake formation permissions can only be granted to tagged resourcese. For more information, see Amazon DataZone updates to AWS managed policies.	June 14, 2024
AmazonDataZoneFullAccess - policy update	Policy update to the **AmazonDataZoneFullAccess**	June 14, 2024

	that enables the Amazon DataZone management console to create secrets on user's behalf with both domain and project tags. Also including the `ram:ListResourceSharePermissions` action to enable administrations from the domain owner account to view the account association status of the associated accounts. For more information, see Amazon DataZone updates to AWS managed policies.	
AmazonDataZoneDomainExecutionRolePolicy - policy update	Policy update to the **AmazonDataZoneDomainExecutionRolePolicy** that adds new APIs to Amazon DataZone that enable users to configure actions for their Amazon DataZone environments. For more information, see Amazon DataZone updates to AWS managed policies.	June 14, 2024
AmazonDataZoneSageMakerProvisioning - new policy	New policy called **AmazonDataZoneSageMakerProvisioning** grants Amazon DataZone the permissions required to interoperate with Amazon SageMaker. For more information, see Amazon DataZone updates to AWS managed policies.	April 30, 2024

AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary - new permissions boundary	New permissions boundary called **AmazonDataZoneSageMakerEnvironmentRolePermissionsBoundary** . When you create an Amazon SageMaker environment via the Amazon DataZone data portal, Amazon DataZone applies this permissions boundary to the IAM roles that are produced during environment creation. The permissions boundary limits the scope of the roles that Amazon DataZone creates and any roles that you add. For more information, see Amazon DataZone updates to AWS managed policies.	April 30, 2024
AmazonDataZoneSageMakerAccess - new policy	New policy called **AmazonDataZoneSageMakerAccess** grants Amazon DataZone the permissions required to grant user access to various resources in the Amazon SageMaker environment. For more information, see Amazon DataZone updates to AWS managed policies.	April 30, 2024

AmazonDataZoneFullAccess - policy update	An update to the **AmazonDataZoneFullAccess** policy that adds access to `DescribeSecurityGroups` action to improve the usability for account administrators configuring blueprints in the console and `GetPolicy` action to help retrieve information about the specified managed policy. For more information, see Amazon DataZone updates to AWS managed policies.	April 30, 2024
AmazonDataZoneS3Manage-<region>-<domainId> - new role	New role called **AmazonDataZoneS3Manage-<region>-<domainId>** that is used when Amazon DataZone calls AWS Lake Formation to register an Amazon Simple Storage Service (Amazon S3) location. AWS Lake Formation assumes this role when accessing the data in that location. For more information, see Amazon DataZone updates to AWS managed policies.	April 1, 2024

AmazonDataZoneGlueManageAccessRolePolicy - Policy update	Updated the **AmazonDataZoneGlueManageAccessRolePolicy** to enable support for permissions that allow Amazon DataZone to enable publishing and access grants to data. For more information, see Amazon DataZone updates to AWS managed policies.	April 1, 2024
AmazonDataZoneDomainExecutionRolePolicy and AmazonDataZoneFullUserAccess - Policy update	Updated the **AmazonDataZoneDomainExecutionRolePolicy** and **AmazonDataZoneFullUserAccess** to enable support for the `CancelMetadataGenerationRun` API. For more information, see Amazon DataZone updates to AWS managed policies.	March 29, 2024
AmazonDataZoneFullAccess - Policy update	Updated the `AmazonDataZoneFullAccess` to enable users to choose their secrets, clusters, vpc's, and subnets in the Amazon DataZone management console rather than type them in a text box. For more information, see Amazon DataZone updates to AWS managed policies.	March 13, 2024

AmazonDataZoneDomainExecutionRolePolicy - Policy update	Updated the **AmazonDataZoneDomainExecutionRolePolicy** to enable support for the ListEnvironmentBlueprintConfigurationSummaries API that is required for creating environment profiles by identifying which blueprints are enabled in which account and region. For more information, see Amazon DataZone updates to AWS managed policies.	February 1, 2024
AmazonDataZoneGlueManageAccessRolePolicy - Policy update	Updated the **AmazonDataZoneGlueManageAccessRolePolicy** to enable support for the AWS Lake Formation hybrid mode. For more information, see Amazon DataZone updates to AWS managed policies.	December 14, 2023
AmazonDataZoneFullUserAccess and AmazonDataZoneDomainExecutionRolePolicy - Policy updates	Amazon DataZone updated the **AmazonDataZoneFullUserAccess** and the **AmazonDataZoneDomainExecutionRolePolicy** policies to support the generative AI-powered data descriptions feature in Amazon DataZone. For more information, see Amazon DataZone updates to AWS managed policies.	November 28, 2023

AmazonDataZoneEnvironmentRolePermissionsBoundary - Policy update	Amazon DataZone made an update to the **AmazonDataZoneEnvironmentRolePermissionsBoundary** managed policy that consists of an additional `athena:GetQueryResultsStream` permission scoped down with the `ResourceTag` condition. For more information, see Amazon DataZone updates to AWS managed policies.	November 17, 2023
AmazonDataZoneRedshiftManageAccessRolePolicy - Policy update	Amazon DataZone updated the **AmazonDataZoneRedshiftManageAccessRolePolicy** policy by removing the check on organization ID for the `redshift:AssociateDataShareConsumer` action. This enables you to share resource across AWS organizations. For more information, see Amazon DataZone updates to AWS managed policies.	November 16, 2023

AmazonDataZoneFullUserAccess - Policy update	Amazon DataZone updated the **AmazonDataZoneFullUserAccess** policy that grants full access to Amazon DataZone, but it does not allow the management of domains, users, or associated accounts .For more information, see Amazon DataZone updates to AWS managed policies.	October 2, 2023
AmazonDataZonePreviewConsoleFullAccess - policy deprecated	Amazon DataZone deprecated the **AmazonDataZonePreviewConsoleFullAccess**.For more information, see Amazon DataZone updates to AWS managed policies.	September 29, 2023
AmazonDataZonePortalFullAccessPolicy - policy deprecated	Amazon DataZone deprecated the **AmazonDataZonePortalFullAccessPolicy**.For more information, see Amazon DataZone updates to AWS managed policies.	September 29, 2023

AmazonDataZoneDomainExecutionRolePolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneDomainExecutionRolePolicy**. This is the default policy for the Amazon DataZone `AmazonDataZoneDomainExecutionRole` service role. This role is used by Amazon DataZone to catalog, discover, govern, share, and analyze data in the Amazon DataZone domain. You can attach the `AmazonDataZoneDomainExecutionRolePolicy` policy to your `AmazonDataZoneDomainExecutionRole` . For more information, see Amazon DataZone updates to AWS managed policies.	September 25, 2023
AmazonDataZoneCrossAccountAdmin - New policy	Amazon DataZone added a new policy called **AmazonDataZoneCrossAccountAdmin** that enables users to work with Amazon DataZone and its associated accounts. For more information, see Amazon DataZone updates to AWS managed policies.	September 19, 2023

AmazonDataZoneRedshiftManageAccessRolePolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneRedshiftManageAccessRolePolicy** that grants permissions to allow Amazon DataZone to enable publishing and access grants to data. For more information, see Amazon DataZone updates to AWS managed policies.	September 12, 2023
AmazonDataZoneRedshiftGlueProvisioningPolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneRedshiftGlueProvisioningPolicy** that grants Amazon DataZone the permissions required to interoperate with the supported data sources. For more information, see Amazon DataZone updates to AWS managed policies.	September 12, 2023

AmazonDataZoneGlueManageAccessRolePolicy - New policy	Amazon DataZone added a new policy called **AmazonDataZoneGlueManageAccessRolePolicy** grants Amazon DataZone permissions to publish AWS Glue data to the catalog. It also gives Amazon DataZone permissions to grant access or revoke access to AWS Glue published assets in the catalog. For more information, see Amazon DataZone updates to AWS managed policies.	September 12, 2023
AmazonDataZoneFullUserAccess - New policy	Amazon DataZone added a new policy called **AmazonDataZoneFullUserAccess** that grants full access to Amazon DataZone via the data portal. For more information, see Amazon DataZone updates to AWS managed policies.	September 12, 2023
AmazonDataZoneFullAccess - New policy	Amazon DataZone added a new policy called **AmazonDataZoneFullAccess** that provides full access to Amazon DataZone via the AWS Management Console. For more information, see Amazon DataZone updates to AWS managed policies.	September 12, 2023

AmazonDataZoneEnvironmentRolePermissionsBoundary - New policy	Amazon DataZone added a new policy called **AmazonDataZoneEnvironmentRolePermissionsBoundary** that limits the provisioned IAM principal to which it is attached. For more information, see Amazon DataZone updates to AWS managed policies.	September 12, 2023
Managed policy update	Updates to the AmazonDataZonePreviewConsoleFullAccess managed policy. For more information, see Amazon DataZone updates to AWS managed policies.	June 13, 2023
Managed policy update	Updates to the AmazonDataZoneProjectDeploymentPermissionsBoundary managed policy. For more information, see Amazon DataZone updates to AWS managed policies.	April 3, 2023
???	Initial release of the Amazon DataZone (Preview) User Guide.	March 29, 2023